User Guide - Ecava IntegraXor 4.1

User Guide - Ecava IntegraXor 4.1

Published 20 Sep 2013
Copyright © 2007~2013 Ecava Sdn. Bhd. All Rights Reserved.

Information in this document is subject to change without notice and does not represent a commitment on the part of Ecava Sdn Bhd. Ecava provides this document "as is", without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Ecava may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in this manual at any time. This product could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes may be incorporated in new editions of the publication.

Table of Contents

DocumentationInformation vii
1. Introduction 1
    1.1. Support 1
    1.2. System Requirements 2
    1.3. Installation 3
    1.4. System Architecture 8
2. Getting Started/Tutorial 11
    2.1. Project Editor 11
    2.2. Project Template 17
    2.3. Inkscape SAGE 18
3. Server User Interface 25
    3.1. Licensing 27
4. Back End Configuration 33
    4.1. General Information/Convention 33
    4.2. General 34
    4.3. Timer 35
    4.4. IO/Port 36
    4.5. Device 36
    4.6. Tag 38
    4.7. Database 42
    4.8. User 44
    4.9. Alarm 45
    4.10. Script 48
5. Front End Configuration 55
    5.1. Menu 55
    5.2. Scripting and Tag Access 57
    5.3. JSON Syntax 58
    5.4. Number Formating 58
    5.5. Tabulate Data 59
    5.6. Printing 59
    5.7. Alarm Viewer 60
    5.8. Trending 62
    5.9. Multilingual support 68
    5.10. Logical Expression 69
6. Inkscape SAGE 71
    6.1. Get Animation 72
    6.2. Set Animation 74
    6.3. Bar Animation 75
    6.4. Color Animation 76
    6.5. Opacity Animation 80
    6.6. Open Animation 81
    6.7. Rotate Animation 82
    6.8. Tooltip Animation 85
    6.9. Slider Animation 86
    6.10. Zoom Animation 87
    6.11. Script Animation 89
    6.12. Popup Animation 90
    6.13. Text Animation 91

6.14. Clone Animation 92
7. Report 95
7.1. Snapshot Report 96
7.2. Statement Report 101
7.3. Report Toolbar Menu 108
8. Security 111
8.1. Security Role 111
8.2. User Configuration 112
8.3. Protection 113
8.4. Additional Security Configuration 114
9. Communication 117
9.1. Setting Communication Port 117
9.2. Port Monitor 120
9.3. Modbus Troubleshooting for SCADA 122
9.4. Interpreting Communication Message 124
9.5. OPC Troubleshooting for SCADA 126
10. Device Driver 127
10.1. Modbus 127
10.2. EtherNet Industrial Protocol 130
10.3. SQL driver for database integration 131
10.4. Profinet 134
10.5. SNMP 136
10.6. IEC60870-5-101/104 138
11. Multiple Redundancy 141
11.1. Multiple Network Redundancy 141
11.2. Network Redundancy Configuration 142
11.3. Multiple System Redundancy 143
11.4. System Redundancy Configuration 145
12. Project Management 149
12.1. Backing up project 149
12.2. Migrating/upgrading project 150
12.3. Group development 150
12.4. Updating Template 150
12.5. Suggested Supporting Tools 150
12.6. Working with Database 151
12.7. Working with Inkscape 153
12.8. Working with Kompozer 172
13. Known Issues 175
13.1. Inkscape SAGE 175
13.2. Project Editor 178
13.3. Front-end 179
13.4. Miscellaneous 180
Index 181

Documentation Information

This User Guide is written based on IntegraXor Version: 4.1.4363.0 dated 20 Sep 2013. The program supports Microsoft Windows Server 2003, 2008, XP, Vista, 7 & 8, in both 32 bit & 64 bit if applicable.

Please avoid to print this manual to reduce paper usage. Much effort has been done and will continue to ensure the program will be friendly enough so reading through this documentation is unnecessary, but merely for reference when in doubt.

Should a printed copy is needed for project documentation submission, this single page info can be printed instead, as the complete electronic document with exact content in PDF format will always be made available online for download at http://www.integraxor.com/doc/ug.pdf. Similar content also accessible from help menu of IntegraXor configuration tool, i.e. Project Editor. Feel free to get more support by visiting http://www.integraxor.com/support.htm or ask for further information.

Thank you for your effort in protecting the environment.

Chapter 1. Introduction

IntegraXor SCADA is a pure Web Server that developed based on W3C standard compliant technologies. No additional web server is needed so installation is very simple and yet cost effective. This also means no additional client program nor plug-in needed except a modern browser for viewing/accessing the mimic screen. So this is a Browser/Server system rather than ordinary Client/Server system.

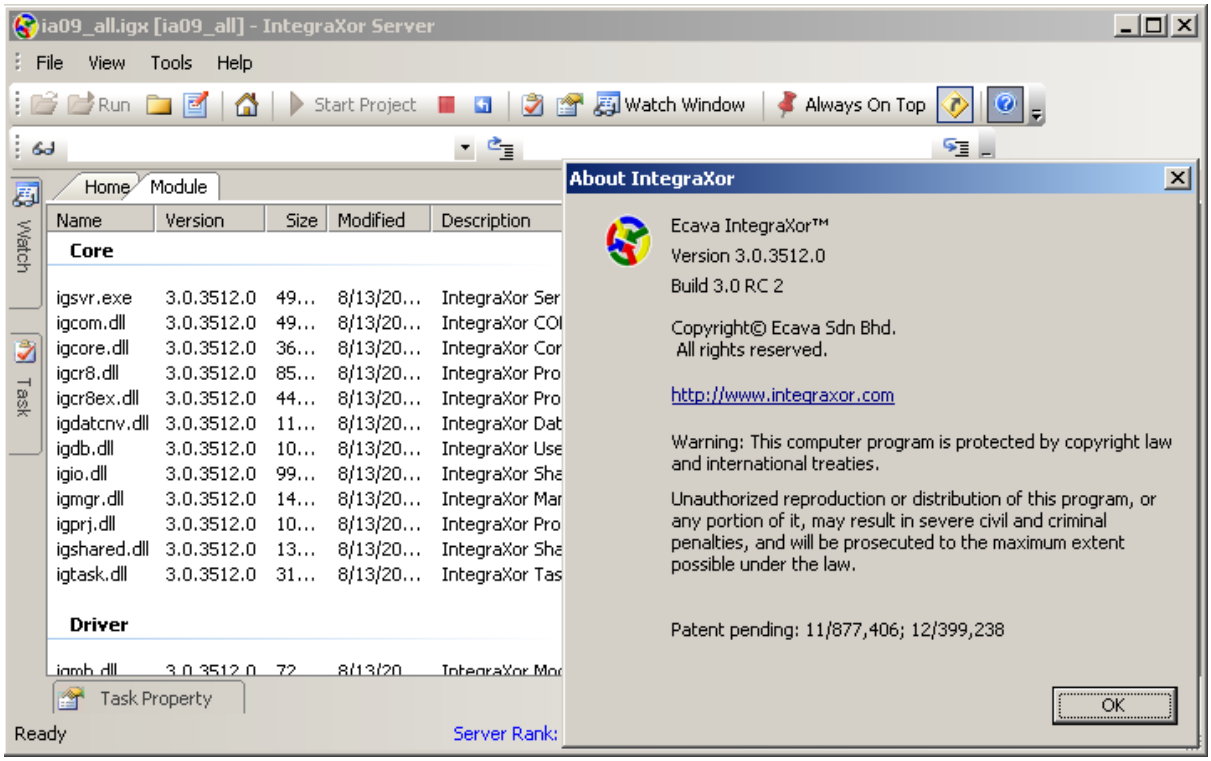
A Getting Started section in this document is provided for users embarking on developing their own project for the first time. The remainder of this document is sectioned for easy reference when doing detailed configuration of your project and graphics.

1.1. Support

A lot of efforts have been put to ensure IntegraXor development tools are user-friendly and robust yet flexible. For additional support or questions, the following options are available.

- IntegraXor Support Forum - http://www.integraxor.com/forum/
- IntegraXor SCADA Developer Network (Blog) - http://www.integraxor.com/blog/
- Email.
- Online chat. (paid customer only.)
- Telephone. (paid customer only.)

Remember to furnish the IntegraXor version and also browser name and version while contacting our support member. Then tell your *expectation* & *steps* involved in producing your problem. Avoid using ambiguous term like "it doesn't work", "it's unpredictable".


ia09_all.igx [ia09_all] - IntegraXor Server
File View Tools Help
Run Start Project Watch Window Always On Top
Home Module
Name Version Size Modified Description
Core
igsvr.exe 3.0.3512.0 49... 8/13/20... IntegraXor Ser
igcom.dll 3.0.3512.0 49... 8/13/20... IntegraXor COI
igcore.dll 3.0.3512.0 36... 8/13/20... IntegraXor Cor
igcr8.dll 3.0.3512.0 85... 8/13/20... IntegraXor Pro
igcr8ex.dll 3.0.3512.0 44... 8/13/20... IntegraXor Pro
igdatcnv.dll 3.0.3512.0 11... 8/13/20... IntegraXor Dat
igdb.dll 3.0.3512.0 10... 8/13/20... IntegraXor Use
igio.dll 3.0.3512.0 99... 8/13/20... IntegraXor Sha
igmgr.dll 3.0.3512.0 14... 8/13/20... IntegraXor Mar
igprj.dll 3.0.3512.0 10... 8/13/20... IntegraXor Pro
igshared.dll 3.0.3512.0 13... 8/13/20... IntegraXor Sha
igtask.dll 3.0.3512.0 31... 8/13/20... IntegraXor Tas
Driver
Task Property
Ready
Server Rank:
About IntegraXor
Ecava IntegraXor™
Version 3.0.3512.0
Build 3.0 RC 2
Copyright© Ecava Sdn Bhd.
All rights reserved.
http://www.integraxor.com
Warning: This computer program is protected by copyright law and international treaties.
Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under the law.
Patent pending: 11/877,406; 12/399,238
OK

Check for IntegraXor version from Help menu - About IntegraXor

It's even better to compress the project folder and attach together with the question if there's no confidential issue. Otherwise a new small project should be reproduced to demontrate the problem to the support member. However, the project folder contains JavaScript or VBScript file which will be blocked by mail server. As such a trailing underscore need to be added to bypass the mail server's extension check. For instance, the compressed file *.zip, *.cab, *.7z etc can be renamed to *.zip_, *.cab_, *.7z_ respectively.

A picture means a thousand words, it's always good to attach a screenshot [http://www.integraxor.com/blog/the-best-way-to-produce-screen-shot] of the problem so the support personnel could understand and identify the root of the problem easily and response faster.

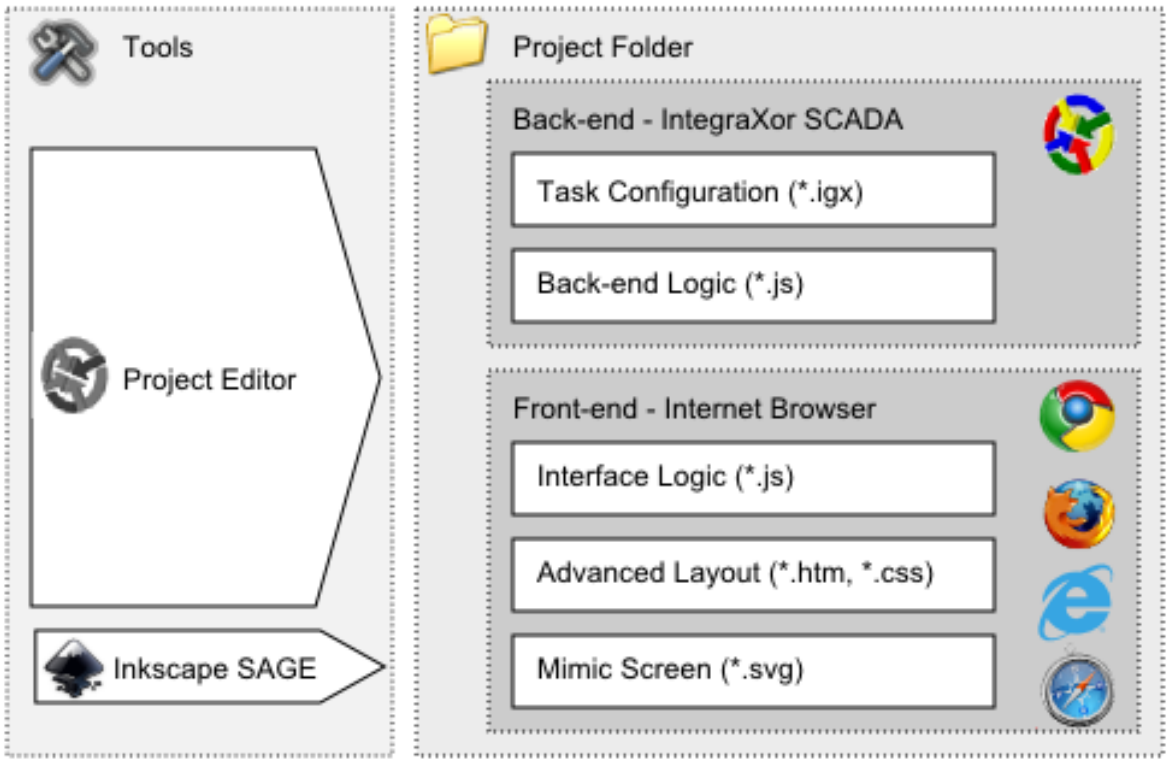
Do not mix old problem with the new one. Always start a new mail for new issue with a descriptive title, most importantly write the new mail to general support email address but not the previous support personnel who wrote to you. This shall ensures faster response [http://www.integraxor.com/blog/how-to-get-faster-response-from-technical-support] as whoever available will reply to you first.

Tip

Pressing F1 twice will copy the version string into clipboard so you can paste straight into your mail content to support.

1.2. System Requirements

Development of a SCADA project is similar to website development. The developer could use various tools to build the website/project. Basic IntegraXor system requires a server for the back-end engine (with built-in webserver) to publish the content and a web browser as client on a standalone system or any networked environment.



Relationship between tools and project

Below is a list of requirements sectionized into Runtime and Development.

Runtime

- IntegraXor server can be installed on Windows XP, Vista or Windows 7.
- Modern Web Browser (which equipped with SVG support) is required for viewing as a client. Mozilla Firefox 3.5+, Google Chrome 3.0+, or Microsoft Internet Explorer 8+ (with Adobe SVG Viewer plugin) is recommended. Adobe SVG Viewer 3.03 can be downloaded at

  http://download.adobe.com/pub/adobe/magic/svgviewer/win/3.x/3.03/en/SVGView.exe.

Development

- Project Editor is included in the main IntegraXor installation file. It is a tool to create and manage the back-end resources such as tags, devices, alarms, users, etc. It also equipped with text editor so user could edit HTML or JavaScript file directly.
- Inkscape SAGE is used for mimic graphic development which can be downloaded at IntegraXor download page http://www.integraxor.com/download.htm. SAGE stands for Scada Animation Graphic Editor.

IntegraXor SCADA server has been designed with efficiency in mind, as such any current basic hardware can be chosen to run the server. And the following 10 years old computer specification has been set as the minimal requirement to run a project with 1024 tags with 5 remote concurrent connections.

Hardware	**Minimum Requirement in XPSP3**
Processor	Intel © Core ™ 2 @ 1.66GHz
Disk Space	~25MB for Runtime + ~200MB for Development
Memory	3 GB

1.3. Installation

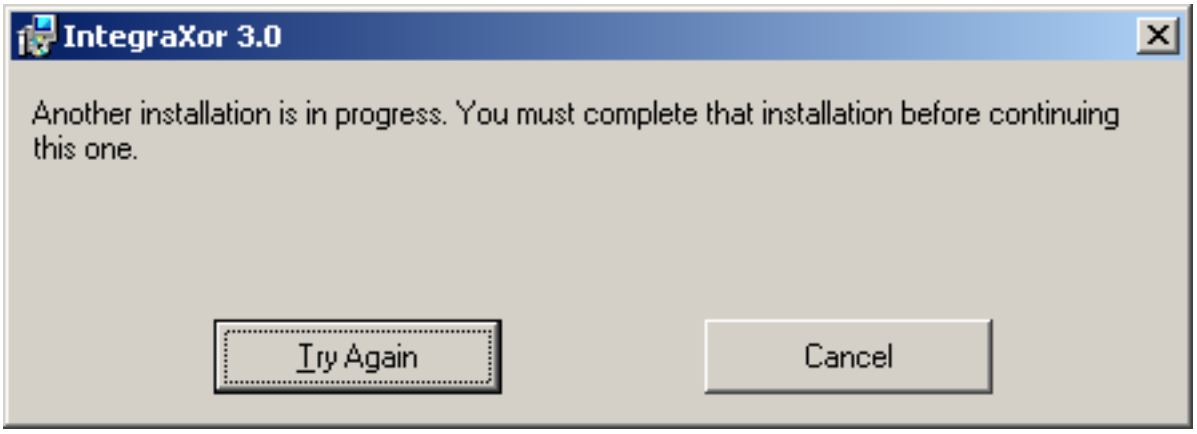
The latest installer can be obtained from http://www.integraxor.com/download.htm. Installing and starting SCADA server requires Administrator Privilege. Please request for Administrator right from system/network admin if installing within company system. However, running browser in remote client will not require such privilege.

Do ensure any installed firewall allowed full internet access for IntegraXor during installation. Otherwise IntegraXor which act as web server will not work. Most of the time, modern firewall shall prompt for permission when IntegraXor is launched for the first time.

Windows Security Alert
To help protect your computer, Windows Firewall has blocked some features of this program.
Do you want to keep blocking this program?
Name: Ecava IntegraXor Server
Publisher: Ecava Sdn Bhd
Keep Blocking
Unblock
Ask Me Later
Windows Firewall has blocked this program from accepting connections from the Internet or a network. If you recognize the program or trust the publisher, you can unblock it. When should I unblock a program?

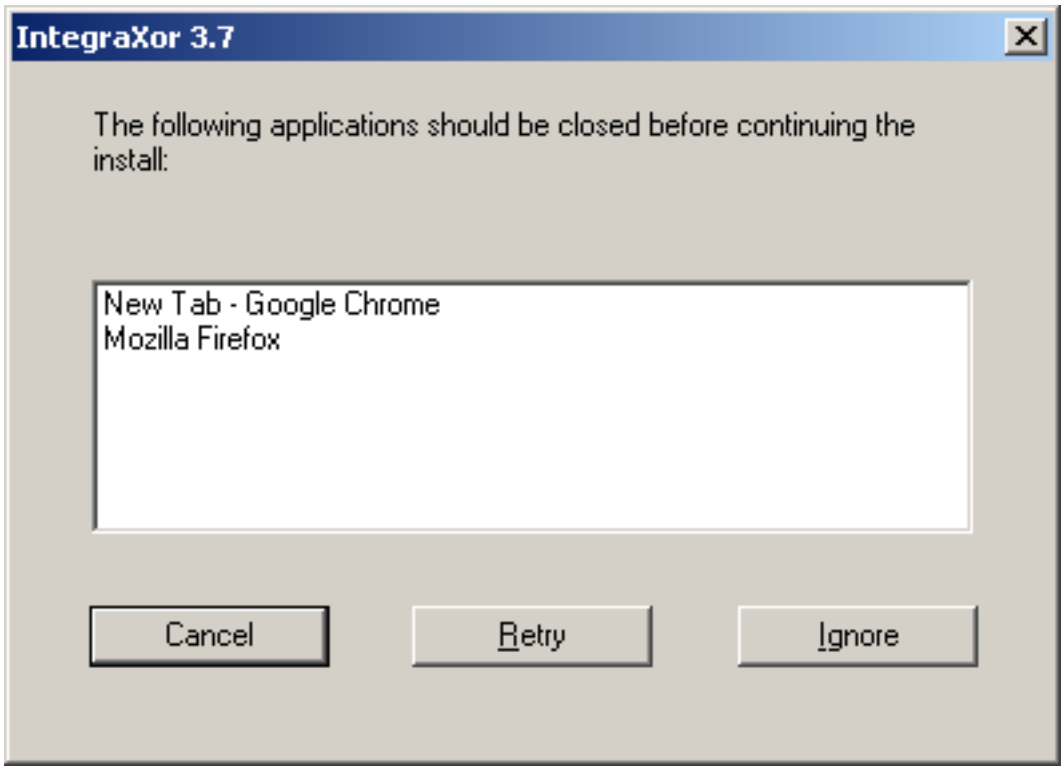
Prompt for firewall security access.

Also ensure no other installation is on going before installing IntegraXor. For instance, Dot Net Framework or some Windows updates could prevent setup from getting initialized. If it's certain that other previous installation is completed or canceled but IntegraXor cannot proceed, check out Task Manager and end any MSIEXEC.EXE process.


IntegraXor 3.0
Another installation is in progress. You must complete that installation before continuing this one.
Try Again
Cancel

Error prompt if other installation or Windows update is on going.

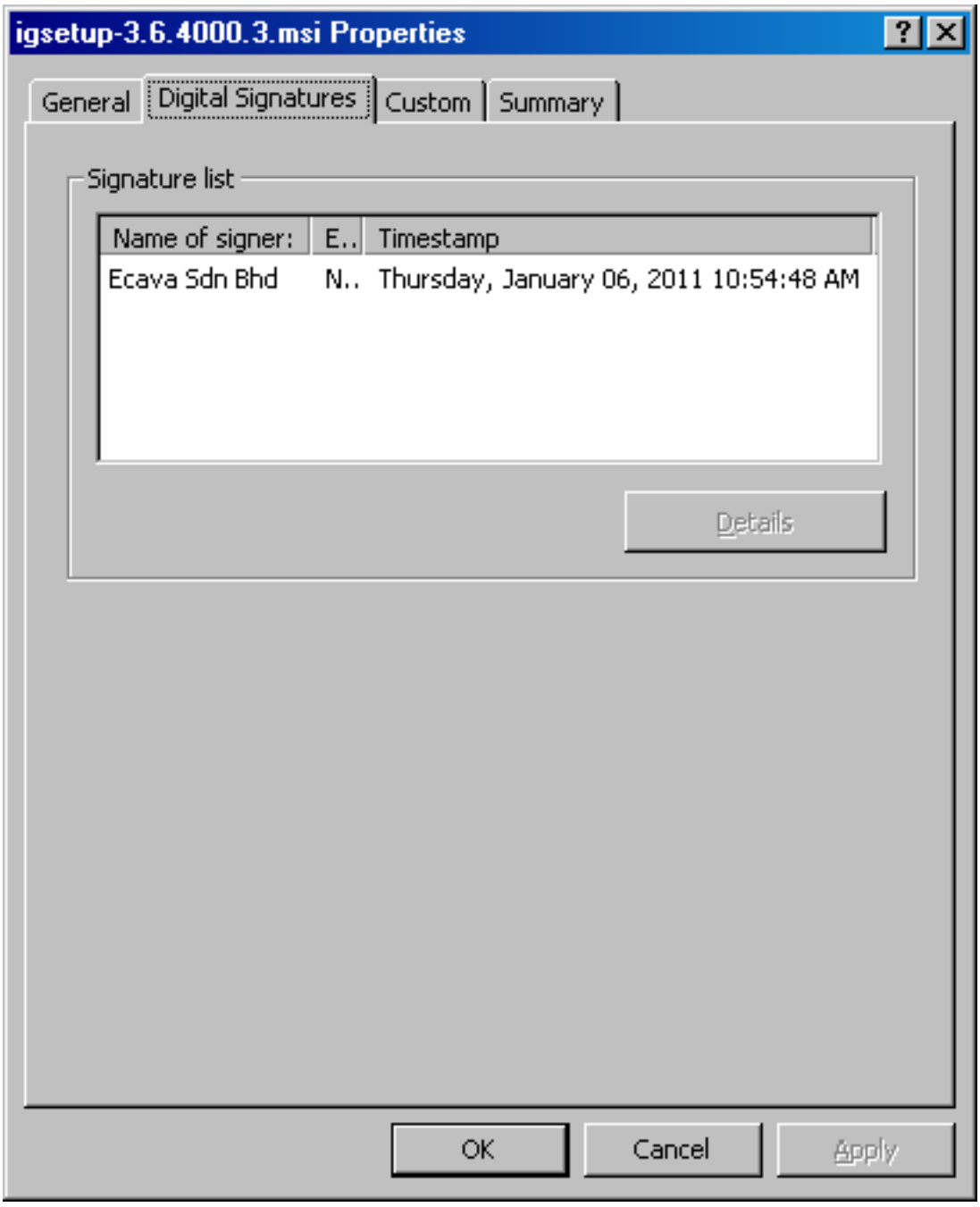
It's not a must to uninstall existing copy for upgrading purpose. However newer installation must be uninstalled when downgrading is performed. And it's pretty common to get false alert from Windows that several application might be sharing the same system library, click "Ignore" to proceed.



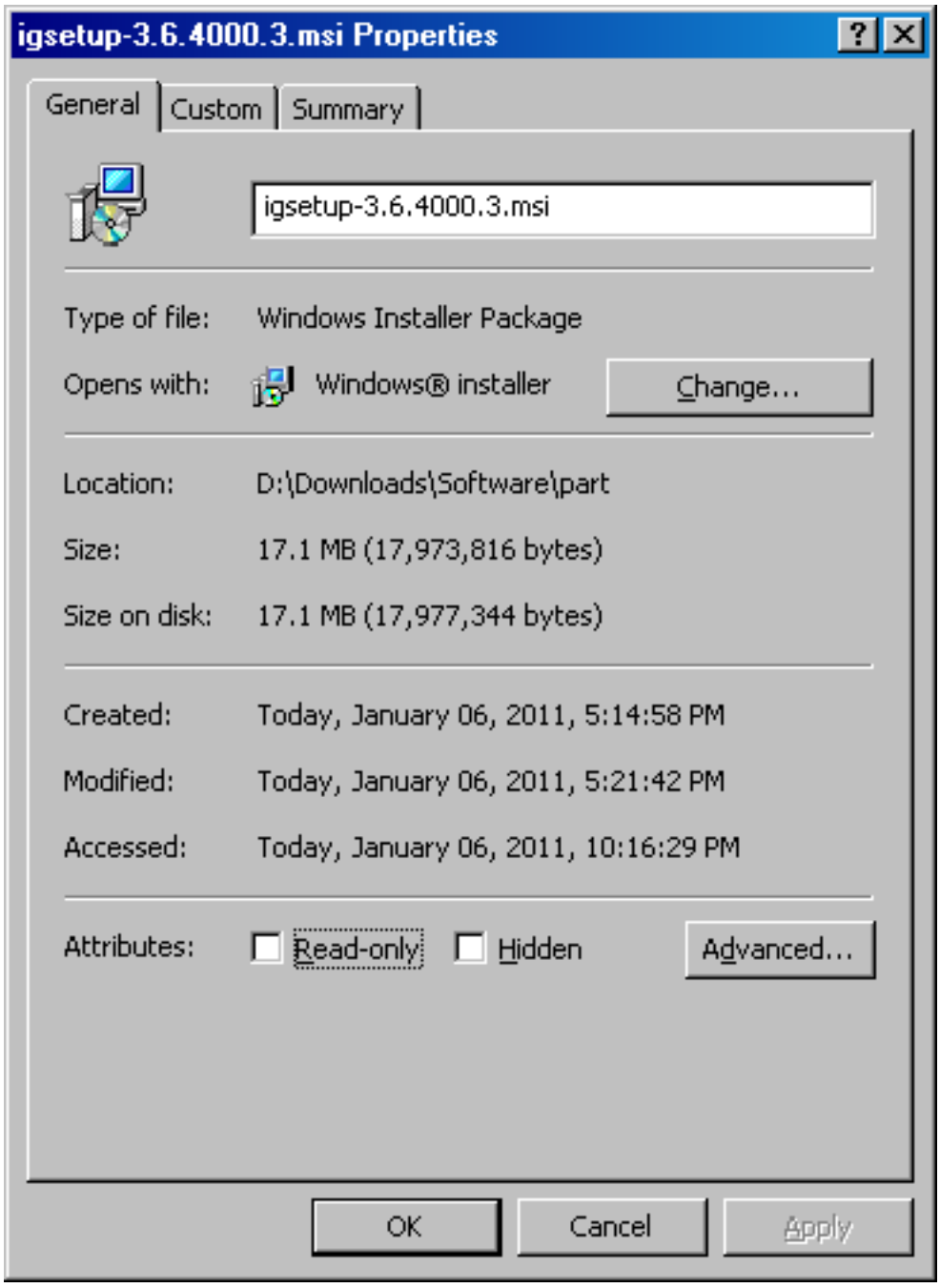
False alert that can be ignored during unistallation.

1.3.1. Incomplete Download Error

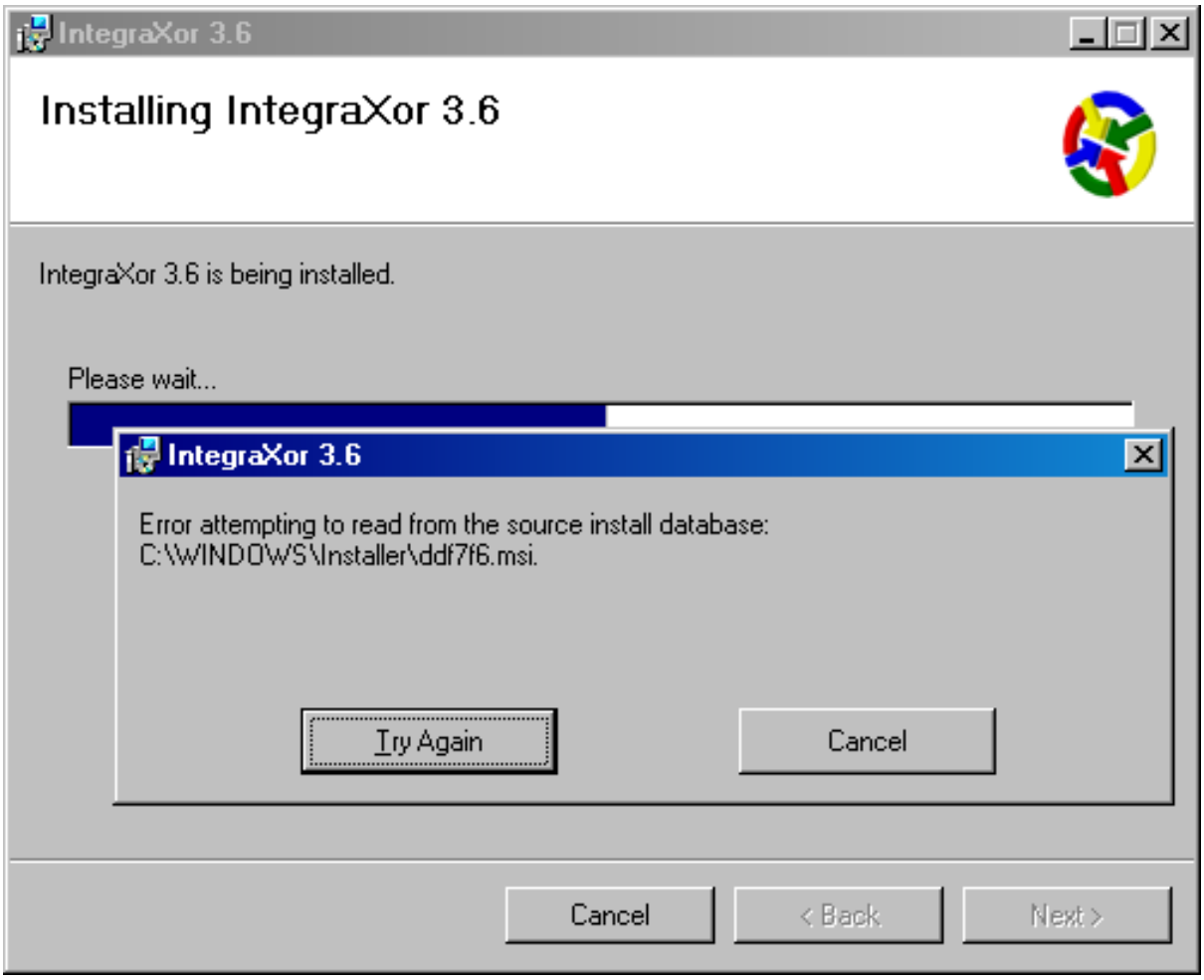
IntegraXor installer is digitally signed before publishing to ensure file integrity. This can be used to check if a file has been tempered or it was not fully downloaded. Below are the file properties window for checking the digital signatures.



"Digital Signatures" can be seen for successful download



"Digital Signatures" tab is missing for incomplete download



Incomplete installer prompts with error: "Error attempting to read from the source install database:"

Tip

Use Free Download Manager [http://www.freedownloadmanager.org/download.htm] for faster and more reliable download.

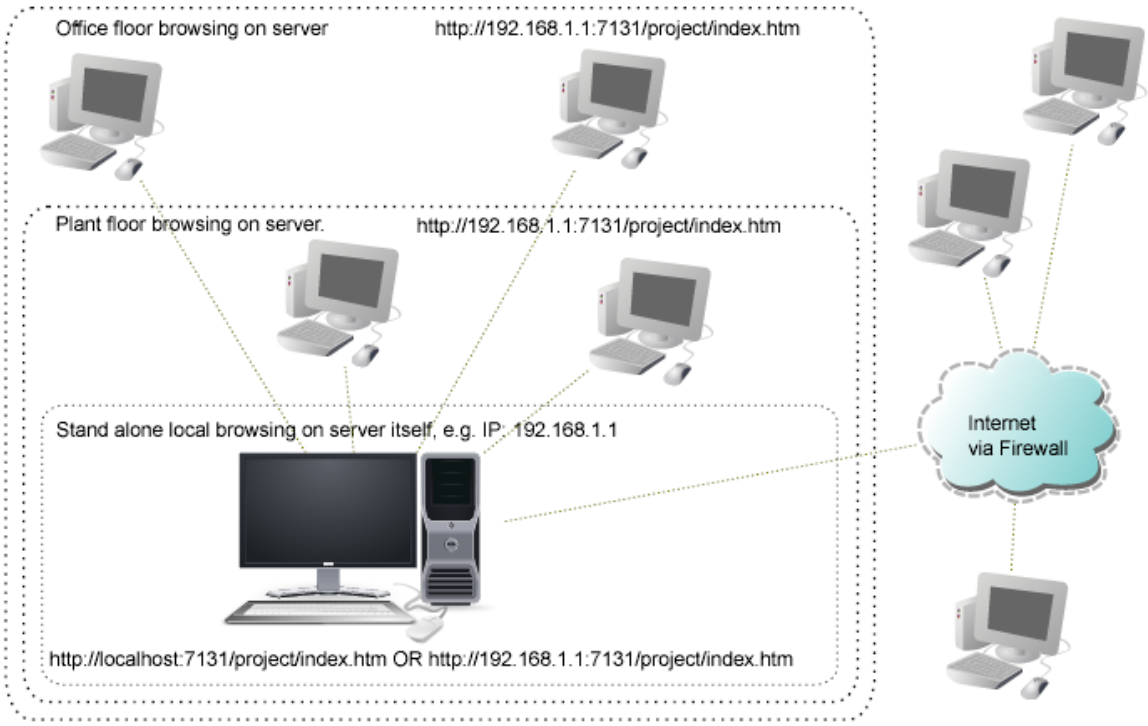
Another possible error may pop up with message:"An error occur during the installation of assembly...". It should be due to corruption in Windows registry, which maybe caused by an unsuccessful install of prior Windows Updates or custom MSI package. This can be confirmed by having similar error while attempt to install another MSI based installer. For instance, 7z compressor [http://www.7-zip.org/download.html]. This issue can be resolved by performing a repair install of Windows to repair any corruption in registry and/or Windows system file, then try to install IntegraXor again.

1.4. System Architecture

IntegraXor project is file oriented and stored in a project folder, including the back-end configuration file (*.igx) and all other HTML & SVG pages. The project can run by simply double clicking the configuration file (*.igx), and can be configured by right clicking then select 'Edit'. Other than *.igx which is the main configuration file for back-end, dblog.mdb is the default MS Access back-end database file for light weight application. The rest of the files are mainly HTML, SVG graphics & scripting files which meant for front-end.

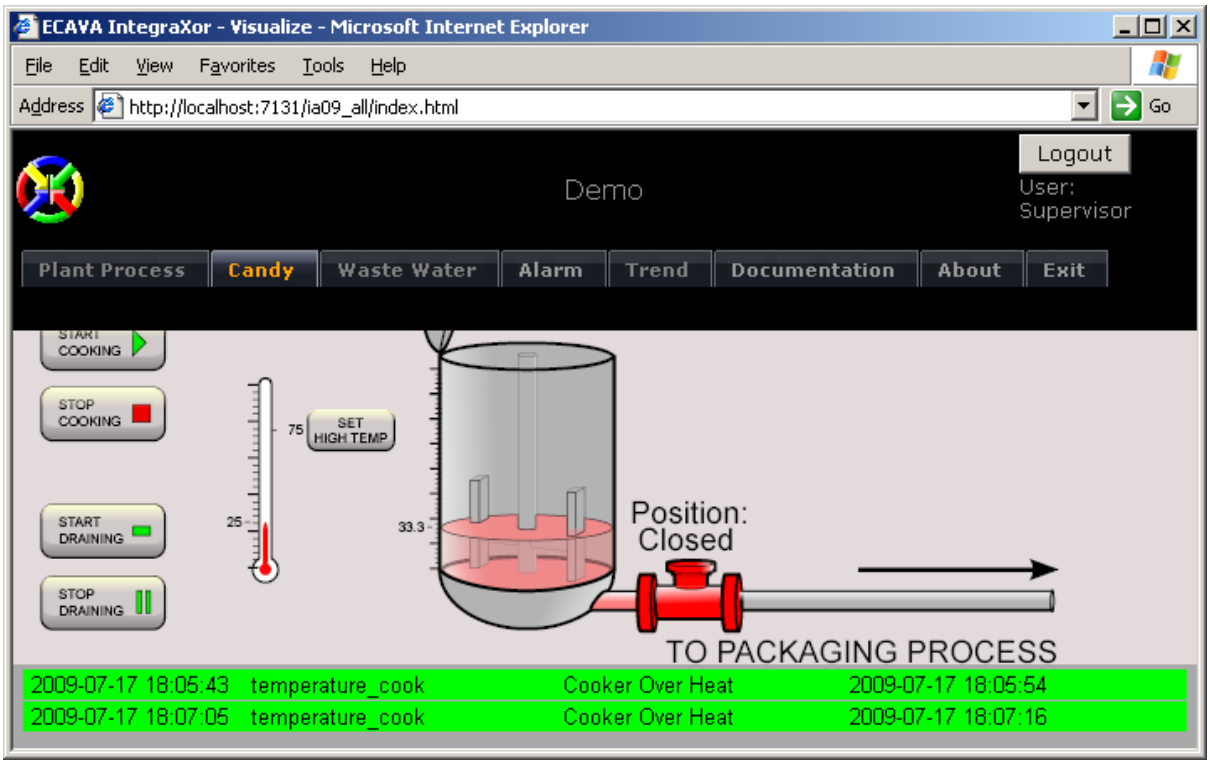
Besides the project root folder, there are several sub folders for better file management. "scripts" for scripting files, "images" for jpg, gif or png raster based images and "styles" for CSS style control. One "system" folder is

completely reserved for front-end engine and shall not be tempered. Just like ordinary web folder, additional folder can be created without restriction. Once the IntegraXor server is running from a project folder, all files can be accessed as if the project folder is a web root folder.



Network architecture on client access via Internet browser.

IntegraXor is designed to listen to port number 7131, this default port number is changeable via Web Server Task. A good example to access a web page shall be typing an address link like this: `"http://localhost:7131/projectID/"`, where the `"projectID"` shall be defined within the Project Editor. For remote access, the URL at remote client shall have the `"localhost"` replaced with the IP address that the IntegraXor is installed. For instance, if IntegraXor is installed at a system with IP address of 192.168.1.1 then the remote client shall have the URL entered as `"http://192.168.1.1:7131/projectID/"`.



A sample project with its URL address.

Tip

The project folder can be moved around without restriction. And Shortcut to *.igx can be placed into Startup Menu to launch the project automatically upon system restart.

Chapter 2. Getting Started /Tutorial

IntegraXor server is a native Web SCADA system. The back-end is a SCADA system which equipped with Web Server and the front-end shall be an ordinary web browser. As such knowledge of web technologies will be in advantage for configuring this system in advanced level. However, many libraries has been created for immediate use without much effort. When new project is selected, a project template will be created, and it's equipped with Menu Toolbar, Graphic Viewer, Alarm Viewer and Report Viewer.

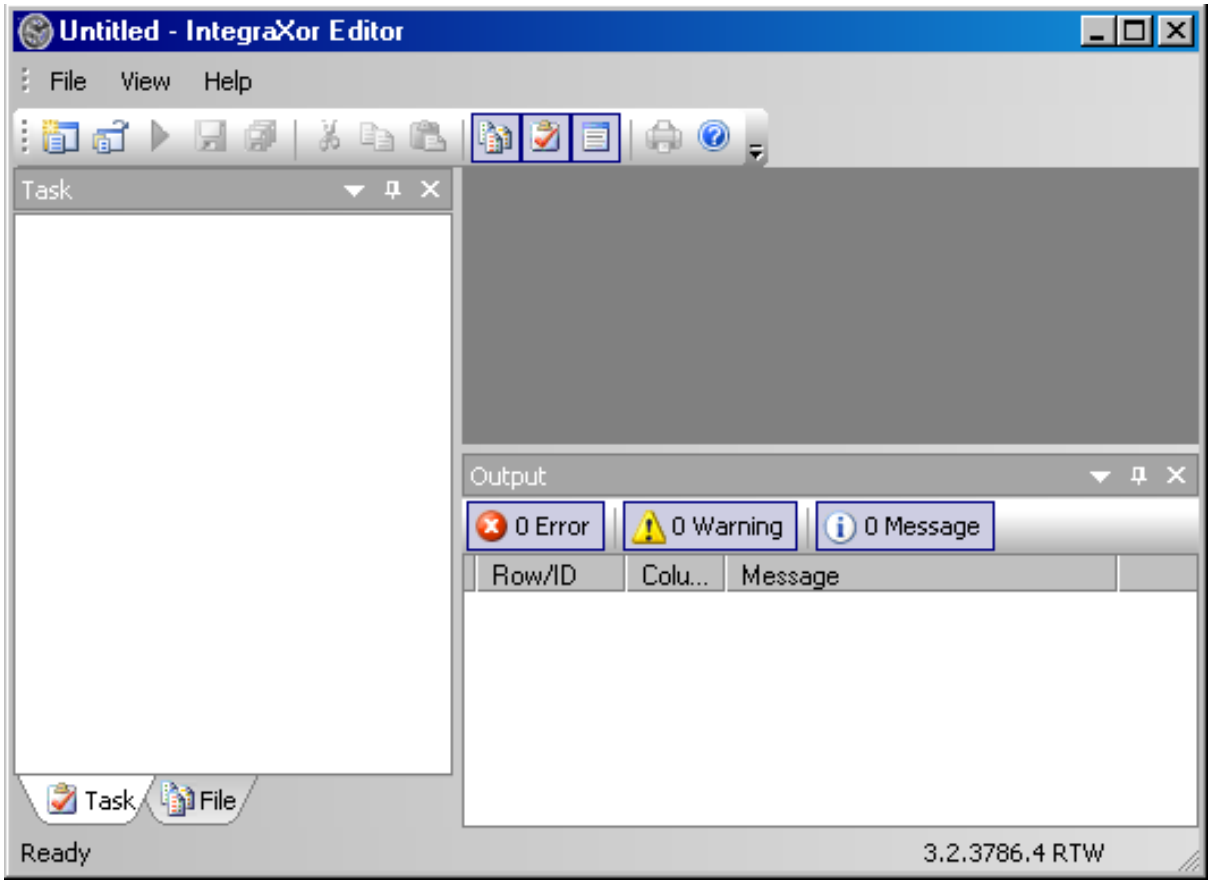
A project template is a folder consists of many files and a few subfolders, however, only one key configuration file *.igx will be found in one project folder. Double clicking this *.igx file shall start the project by launching the IntegraXor server automatically. Right click and select 'Edit' would launch the Project Editor for configuration instead. All mimic screens can be developed by using standard compliant HTML or SVG files, and they can all be added into Menu.

Tip

A SCADA tutorial in separate PDF file is available for download (~3MB) [http://www.integraxor.com/download/tutorial.pdf] and is a good way to get start.

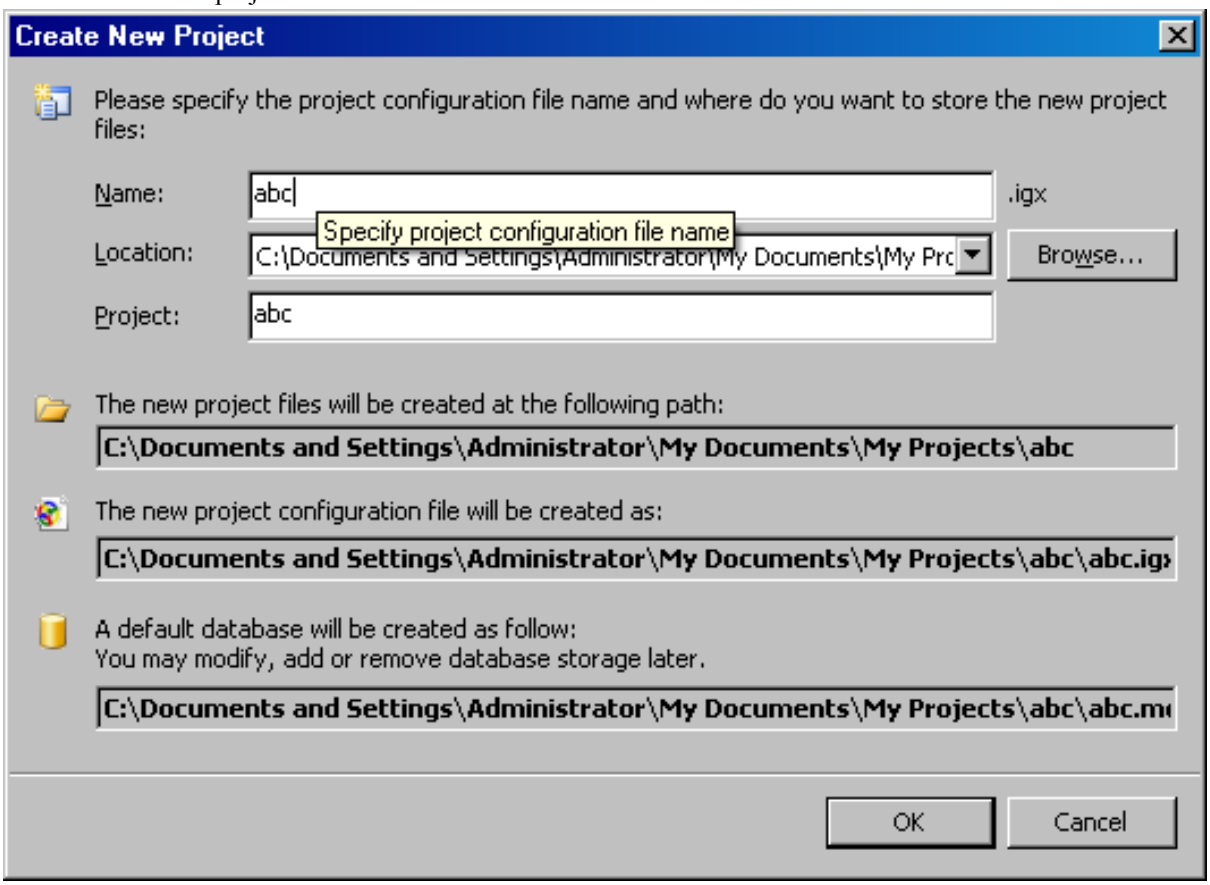
2.1. Project Editor

Project editor is the first and foremost tool needed in order to configure a project. Start Project Editor from system menu, an untitled Project Editor shall load as per below. The Project Editor is split into one hierarchy tree view and its associate tabulate content on the right. And an output window for displaying information after configuration is being validated.



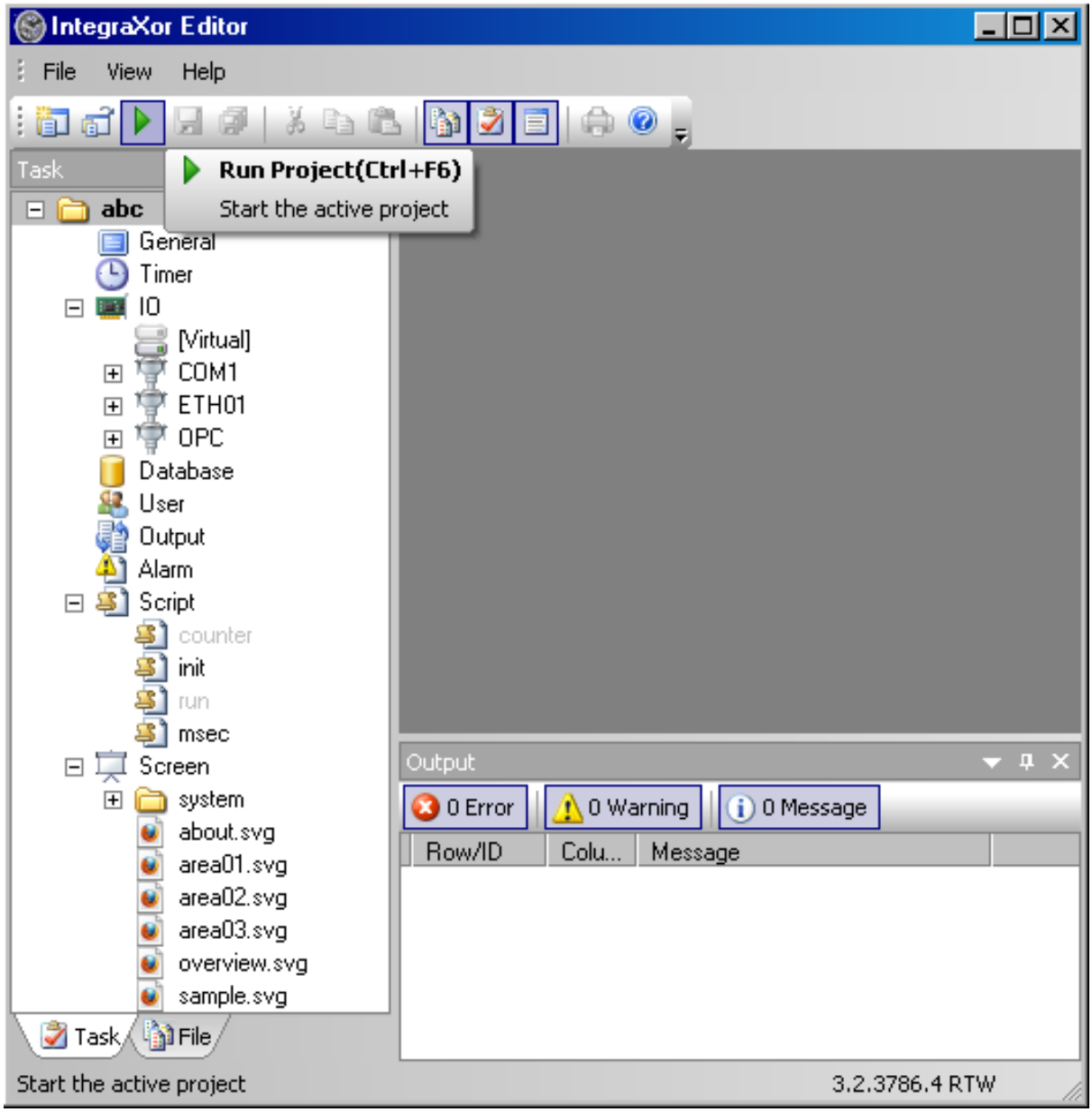
Project Editor screen shot before loading any work.

To start working on a new project, click File-New Project. A pop up prompting a project name shall display as below. This project name will be treated as project ID which will be used through out the whole project. Carefully choose a short and precise name without any space. Only alphabets and numbers are allowed. Press OK to create a new project.


Create New Project
Please specify the project configuration file name and where do you want to store the new project files:
Name:
abc
.igx
Specify project configuration file name
Location:
C:\Documents and Settings\Administrator\My Documents\My Pro
Browse...
Project:
abc
The new project files will be created at the following path:
C:\Documents and Settings\Administrator\My Documents\My Projects\abc
The new project configuration file will be created as:
C:\Documents and Settings\Administrator\My Documents\My Projects\abc\abc.igx
A default database will be created as follow:
You may modify, add or remove database storage later.
C:\Documents and Settings\Administrator\My Documents\My Projects\abc\abc.m
OK
Cancel

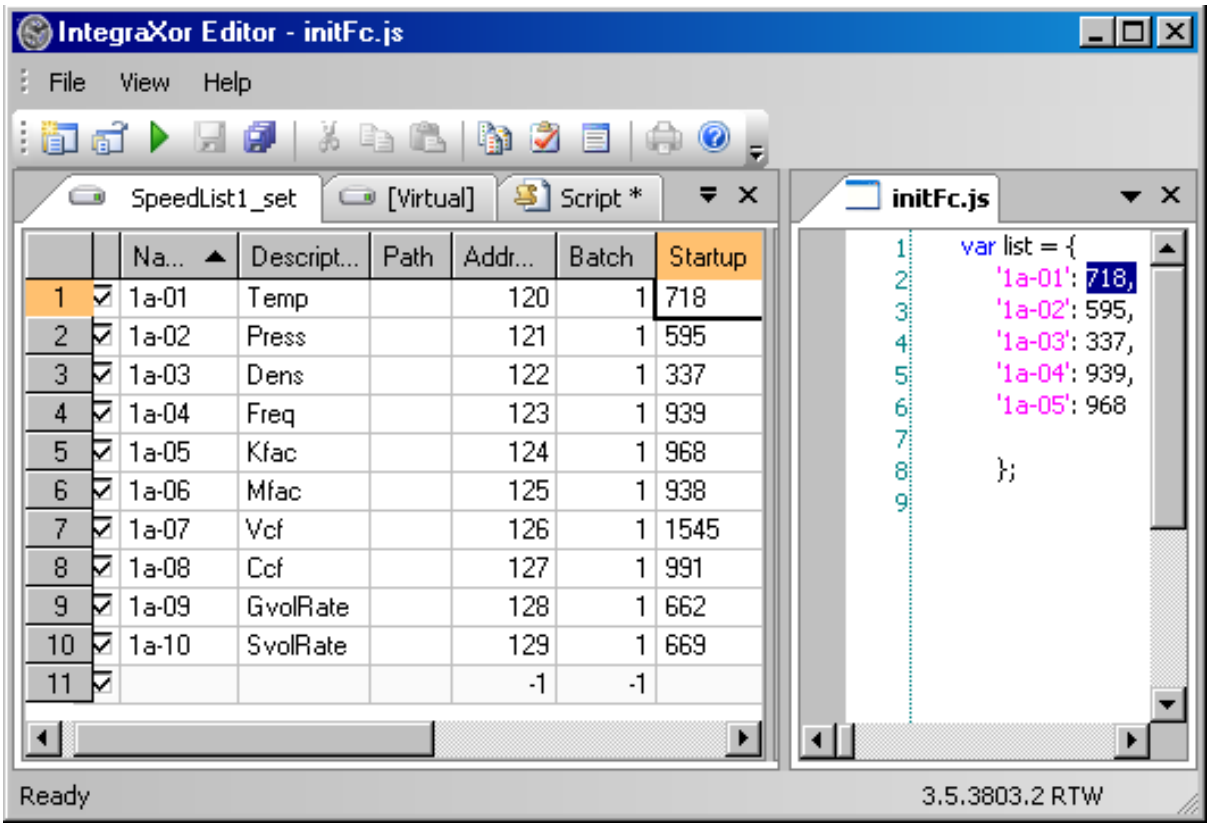
Saving and naming a new project.

Below showing a newly created project comes with basic default configuration which is ready to run. Click on the Run button shall launch the server right away.



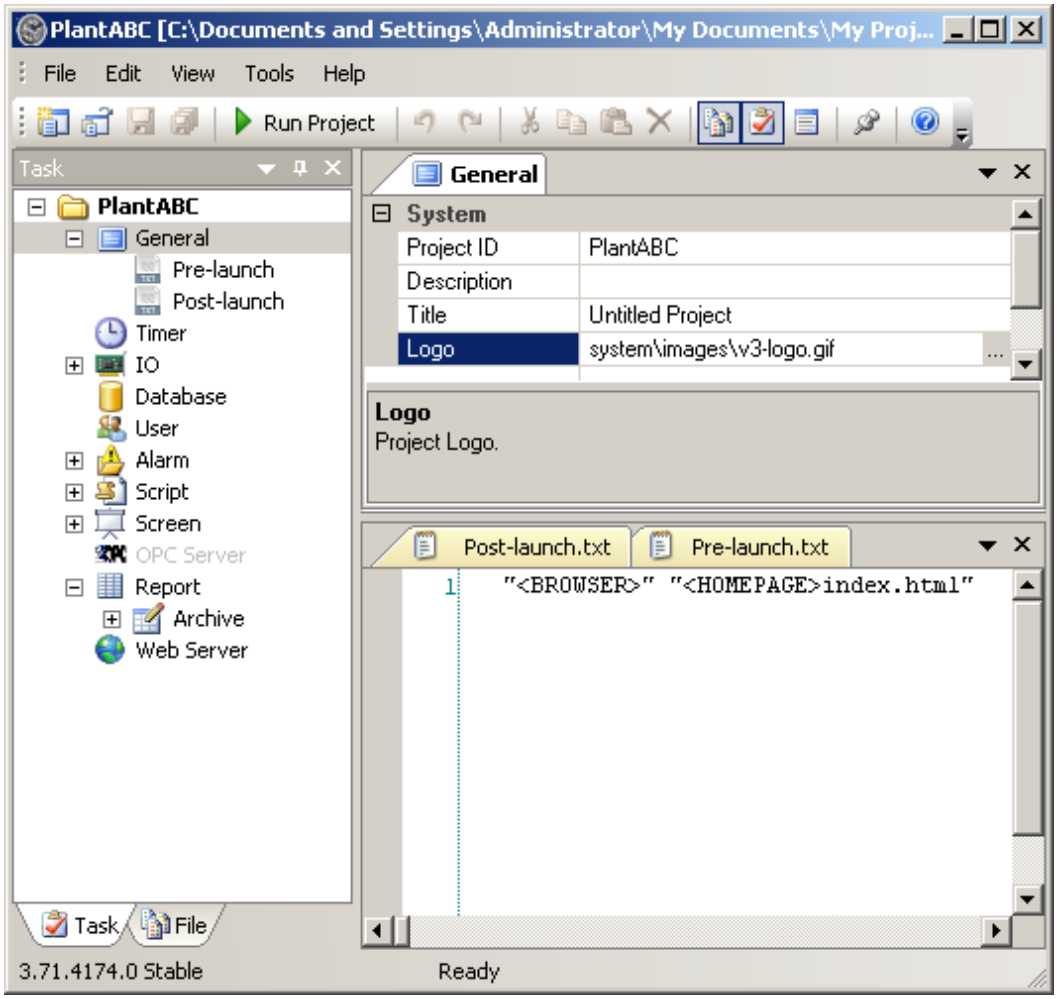
Start running a new project.

All window panes can be reposition by dragging the individual window to any desired layout according to personal preference. And all columns within each Tab can be dragged around to be reposition for better visibility according to frequency of usage. Moreover, all Tabs also can be dragged around for split view for ease of comparison or references.



Project Editor layout or view can be fully customized. Picture showing Tree Menu and File Menu are closed. Then Tag Pane and Script Pane has been placed side by side for ease of reference.

General A default working template will be created, all necessary entries and links shall be entered automatically. For instance, Internet Explorer shall be automatically launched with the associated file path once the project is started. Note that the project ID will also be automatically created based on the new project name entered earlier.



Start running a new project.

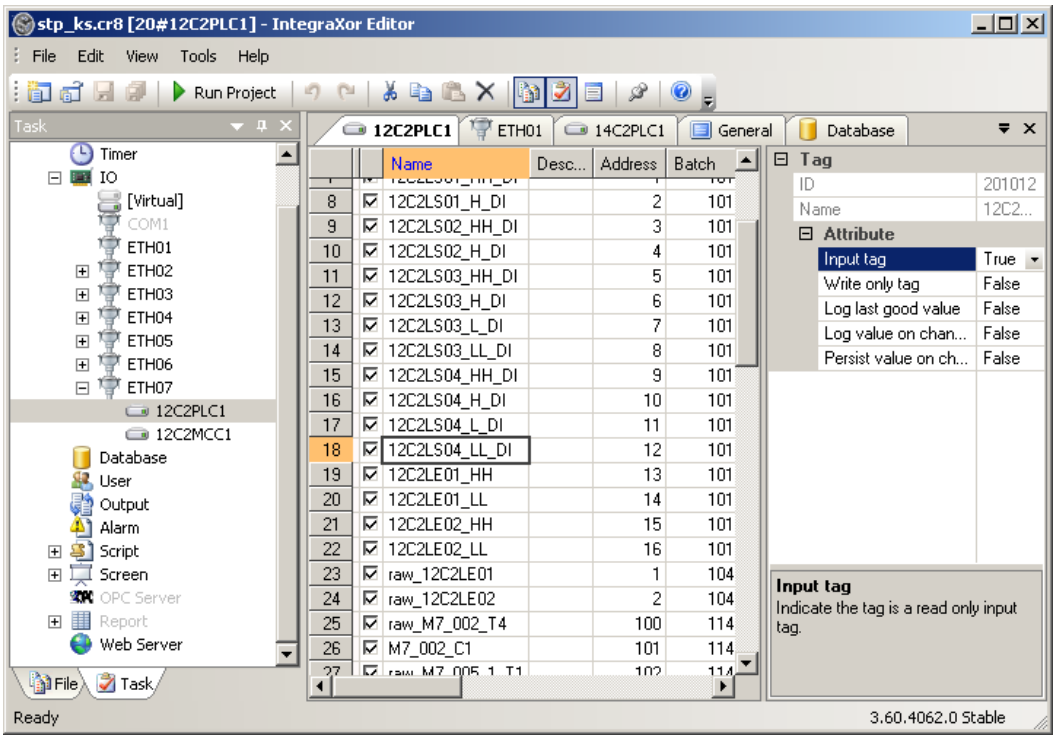
Timer Timer configuration is the engine that triggering the whole project to run. Three of the tasks that required Timer for triggering are Data Logging for trending, Scripting Logic that need to be looped through, and Tag polling from devices. Several default entries have been made to make configuration works easy. For instance, SEC01 will be triggered in every second, SEC02 will be triggered in every two seconds and so on and so forth.

Port Port folder consists of Device folder, Device folder yet consists of Tag table entry. The hierarchy totally illustrated the relationship between Port, Device and Tags. One port may have one or more Devices attached to it, and depending on project size, one Device may have several to thousands of tags attached to it. The template has pre-configured with a COM port and ETH port.

Device One Device has been pre-configured for Modbus Serial protocol. And another Device has been pre-configured for Modbus Ethernet protocol. However, engineer must ensure the port setting is completely match with device's communication setting. There will be always one Virtual Device which connect to nothing but simply allow engineer to enter non I/O tag into it.

Tag The Tag System hierachy is like a tree structure, Tag is under Device, yet Device shall be under Port. As such first step is to determine how is the device(s) being connected to the SCADA server. Is it via Serial Com port or Ethernet Port? Then name a Device under the created port and choose the communication protocol (device driver). After that all need to be done is creating list of tags that associate with the data addresses in the device(s). For any tag that do not need to link to external device, enter as virtual tag under virtual device. To enter a new tag, simply add in the

tag details at the last row.

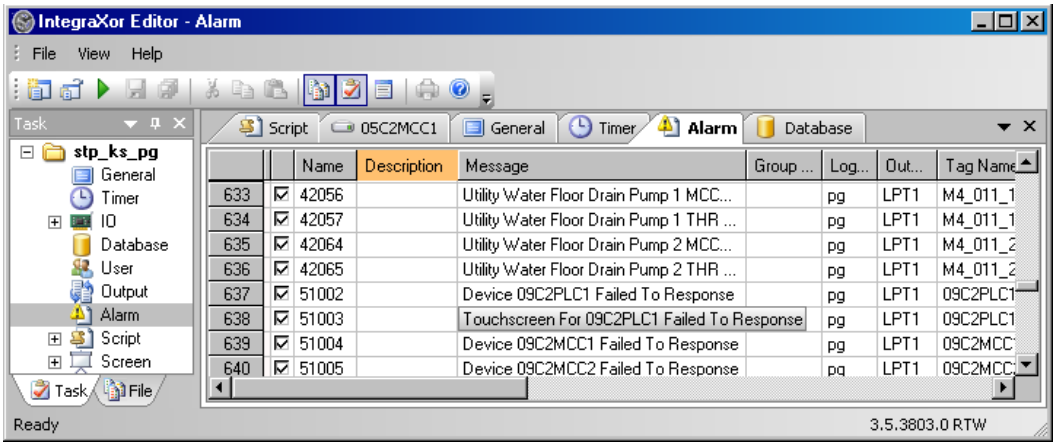


Tag entry into Project Editor.

Database Database is used for logging data for alarm or event, trending or reporting. No additional work required to setup a real time database system, a Microsoft Access database will always added into one project folder. If larger or more reliable database is needed, user could opt for any external ODBC compliant database.

User The default users created shall be sufficient in common applications. These users only needed if security system need to be implemented.

Alarm Enter alarm entry as shown.

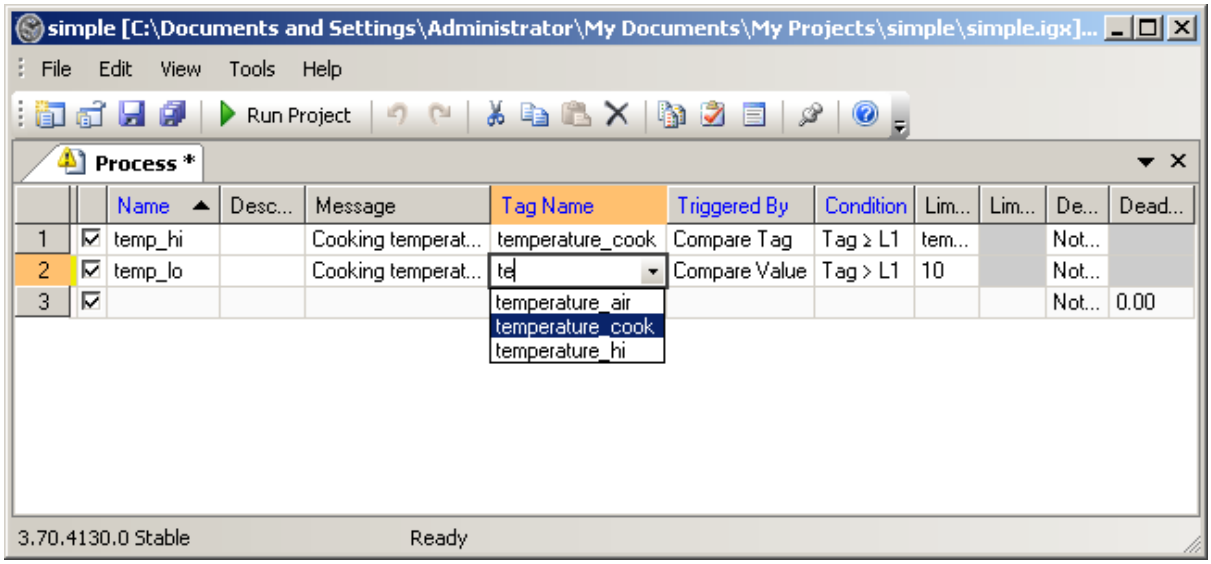


Alarm entry into Project Editor.

Script The scripting language used is base on standard JavaScript. There are a lot of resources on the web to pick up this ubiquitos Internet language, as such this manual will not describe anything on JavaScript. VBScript is also supported if targeted browser is restricted to IE only.

2.1.1. Auto-Complete for Tag Entry

All tag entry field supports auto-complete, user only need to type the first or first few characters to filter out non-relevant tags to speed up entry work.



Example shown typing 'te' will narrow down listing for tags that begin with 'te' only.

2.2. Project Template

Project template will be created when New Project is initialized under Project Editor. The project template folder provides several essential elements for a complete HMI/SCADA system. One and foremost is the server configuration file `*.igx`. The directory structure is very similar to ordinary web site, it could consist sub-directories like 'images', 'scripts', 'styles' and 'system'. And 'system' directory must always present and must not be altered as it consists of other sub-directories that required by IntegraXor.

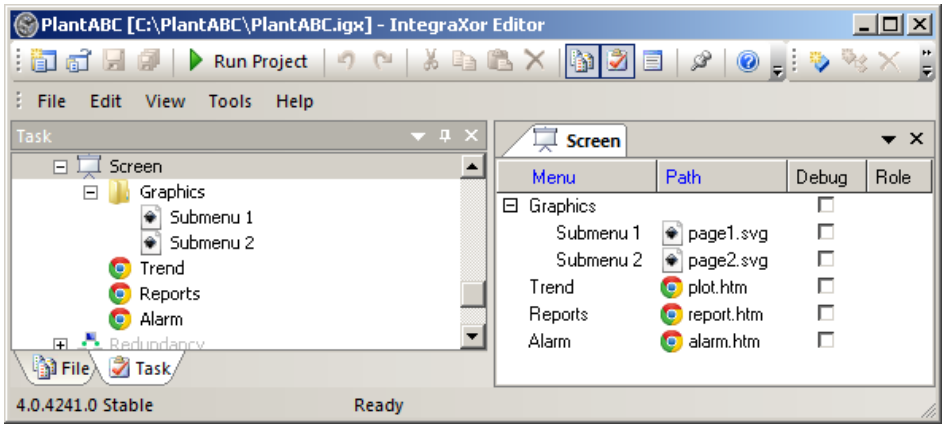
Template is part of IntegraXor system and is released together with server and should be compatible as a whole. Do not mix around system files between different version of template unless the version is stated as compatible.

Layout

Layout of IntegraXor interface consists of 3 parts: Header, Body and Footer. Similar to web surfing experience, header will behave as a link, and body will change content according to header. Footer is general alarm. Technique behind the automation is by using javascript to evaluate a menu setting file, and extract the source from project directory.

Header and Body

The "header" is a small pane for logo placement, project title and current time, more importantly, header is used as menu placement to access all screens. To minimize S.I. Engineer's work, he/she will only need to edit Screen task in project editor. Below is an example:



Menu arrangement.

Above example will generate a main menu and a sub menu of "Graphics". After "Graphics" is being clicked, sub menu of "Page 1" and "Page 2" will appear, and they have their own source file respectively attached at the right side of colon symbol ':'

All the source files should be placed at root directory of project folder.

Note

Currently the supported file types are: SVG, PDF and HTM/HTML files. In addition, description text (page title) at left side of colon symbol ':' must be unique.

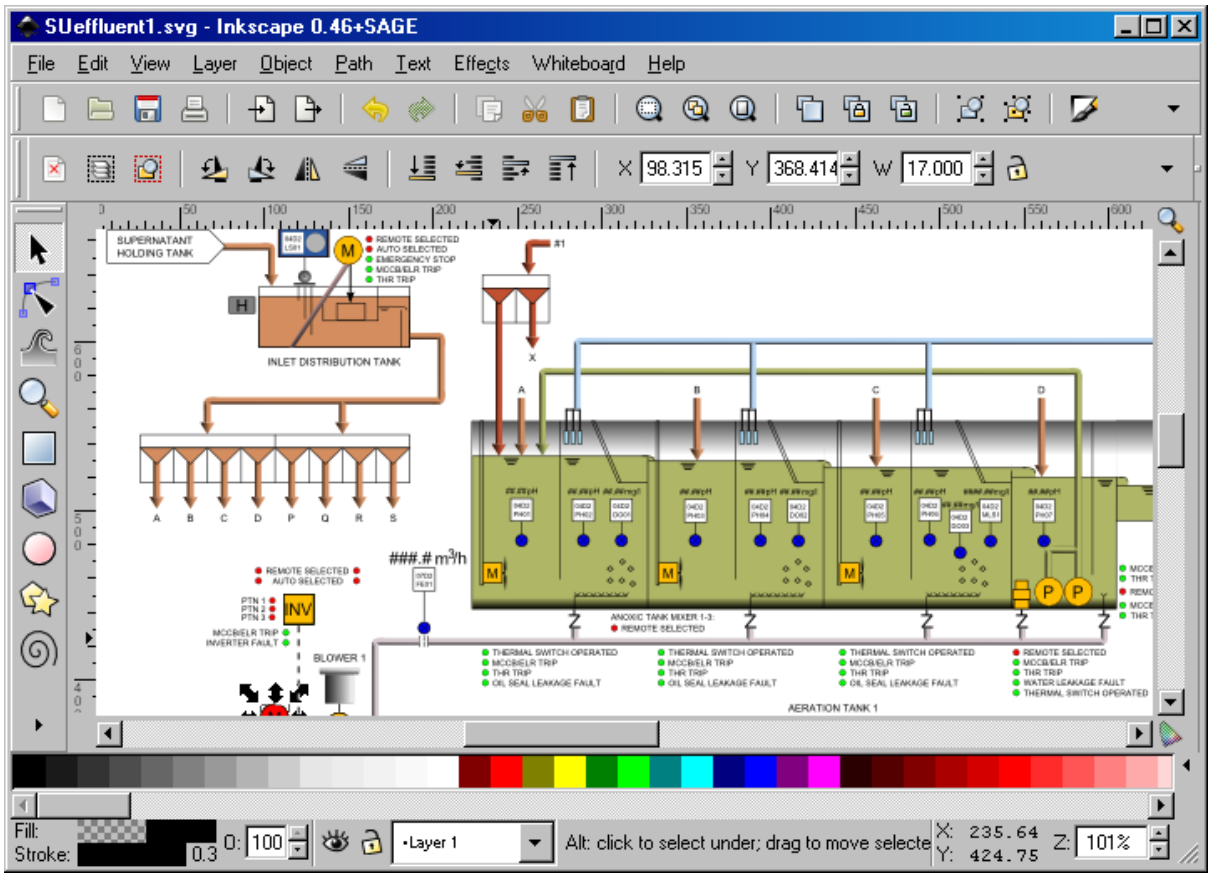
Footer — The "footer" is mainly used for holding the alarm/event bar. Footer can be omitted if end user does not need to have latest alarm displayed at the bottom at all time.

Alarm Viewer — Alarm Viewer is one full page of alarm listing that allows operator to view, acknowledge and track historical alarm. It is part of the default page but can be removed if not needed.

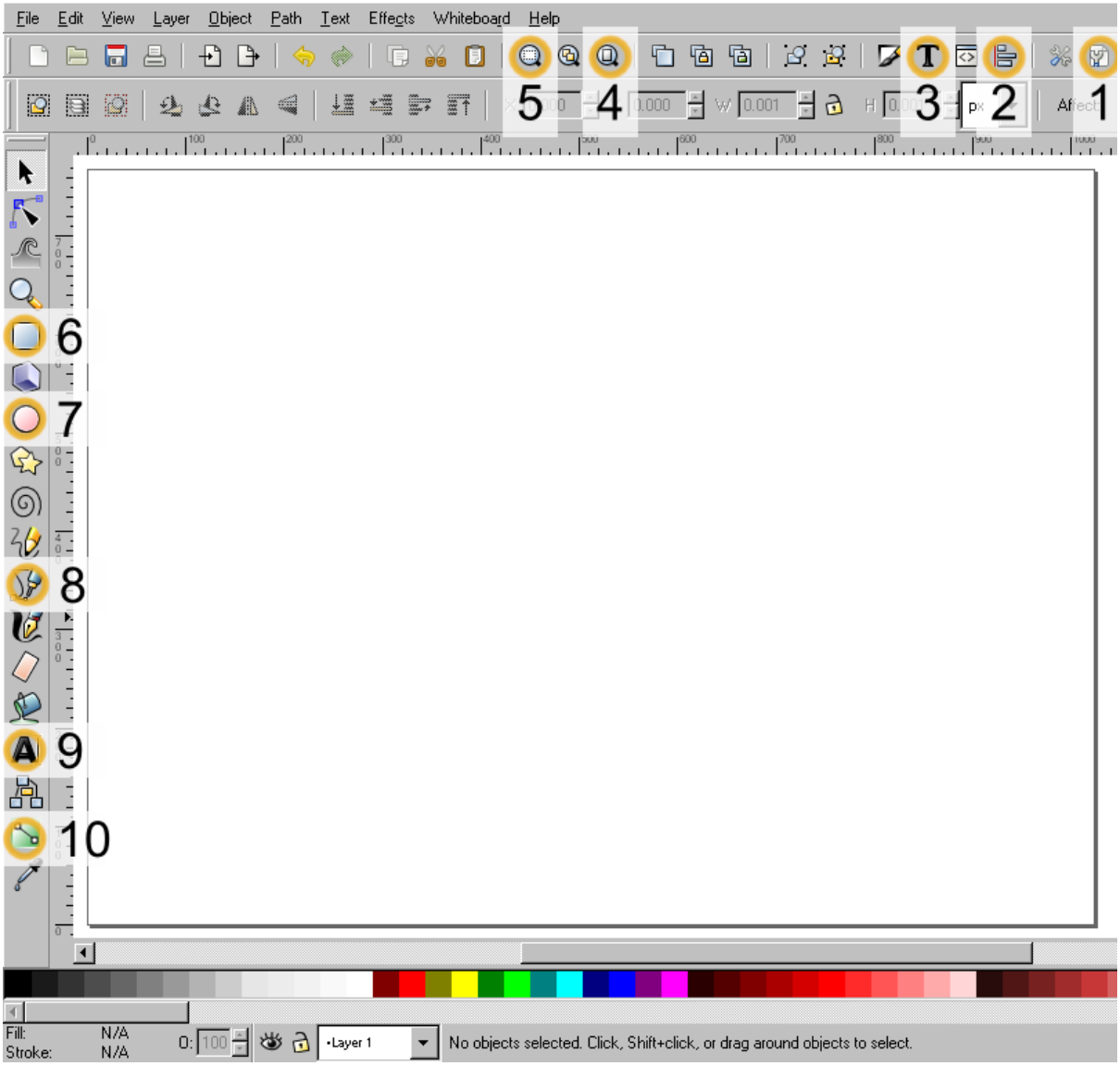
Database — For non-critical database usage, a default Microsoft Access database "dblog.mdb" is distributed for a quick startup. This empty database is ready for use without any intervention.

2.3. Inkscape SAGE

Inkscape is a powerful & open-source SVG editor that can produce very impressive mimic, none of the existing SCADA system could generate anything close to Inkscape. Despite of its rich and powerful set of drawing tools, it's still surprisingly user friendly. There's already a built in tutorial under Help menu in Inkscape for learning it, and many resources also available online. One of them is available at http://tavmjong.free.fr/INKSCAPE/MANUAL/html/. This link can be reached by clicking on Help Menu - Inkscape Manual. Integraxor added a user interface for handling SCADA animation called SCADA Animation Graphic Editor, SAGE.



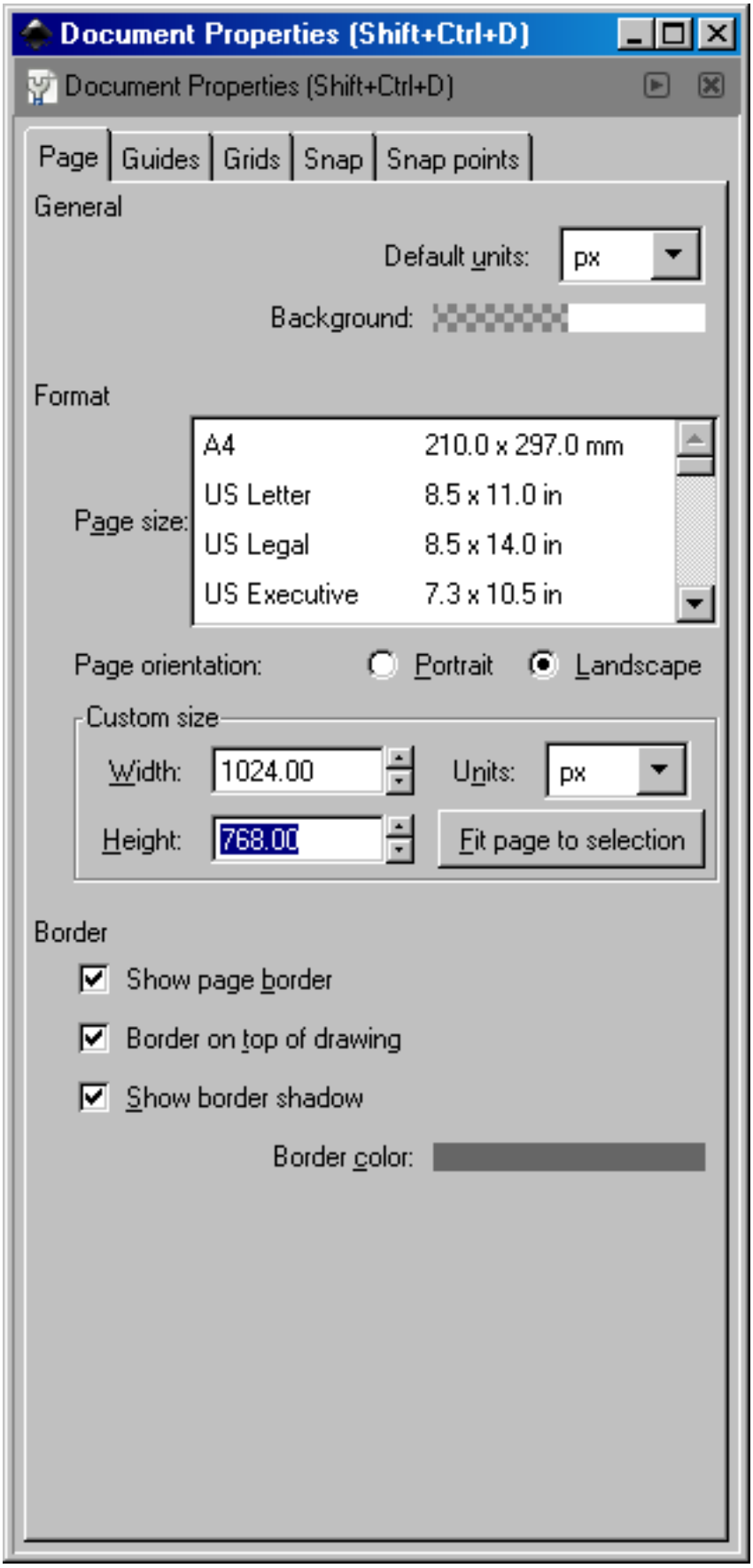
Inkscape graphic editor.
Inkscape is a professional graphic designer level drawing program, only a few sets of drawing tool need to pay attention with in order to draw a SCADA images as per the next section highlights.



Highlight of essential tools.

1. *Document setting*

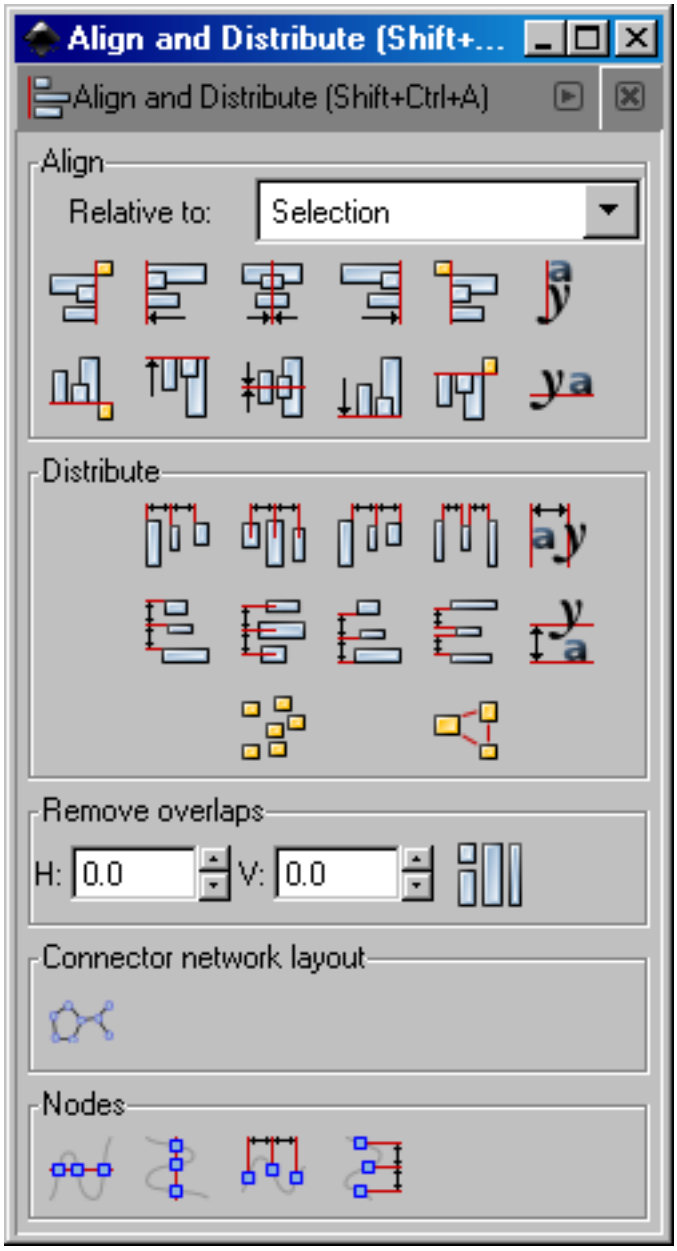
   Normally width is the screen width, and height is slightly smaller then screen height, depending on requirement of header or alarm bar footer. The basic idea is to choose a size that proportional to the final screen resolution, as IntegraXor will fit the mimic to the client screen resolution automatically.



Document properties.

2. *Alignment*

   Alignment is a handy tool to ensure tidiness. However, it's not very much require if engineer turn on the grid snap option, this is again up to personal preference.



   Alignment properties.

3. *Font properties*

   Shortcut key is 'Shift+Ctrl+T'.

4. *Zoom to whole page*

   Shortcut key is 'z' follow by '5'.

5. *Zoom to selection*

   Shortcut key is 'z' follow by '3'.

6. *Draw rectangle*

   Shortcut key is 'r'. Rectangle can be easily tuned to have round or sharp corner.

7. *Draw circle or semisphere (Ellipse/arc)*

   Shortcut key is 'e'.

8. *Draw (Bezier) lines*

   Shortcut key is 'b'. Click from point to point for drawing straight line, drag from point to point for drawing curve.

9. *Insert text*

   Shortcut key is 't'. Simply click and type to enter text to avoid browser support issue.

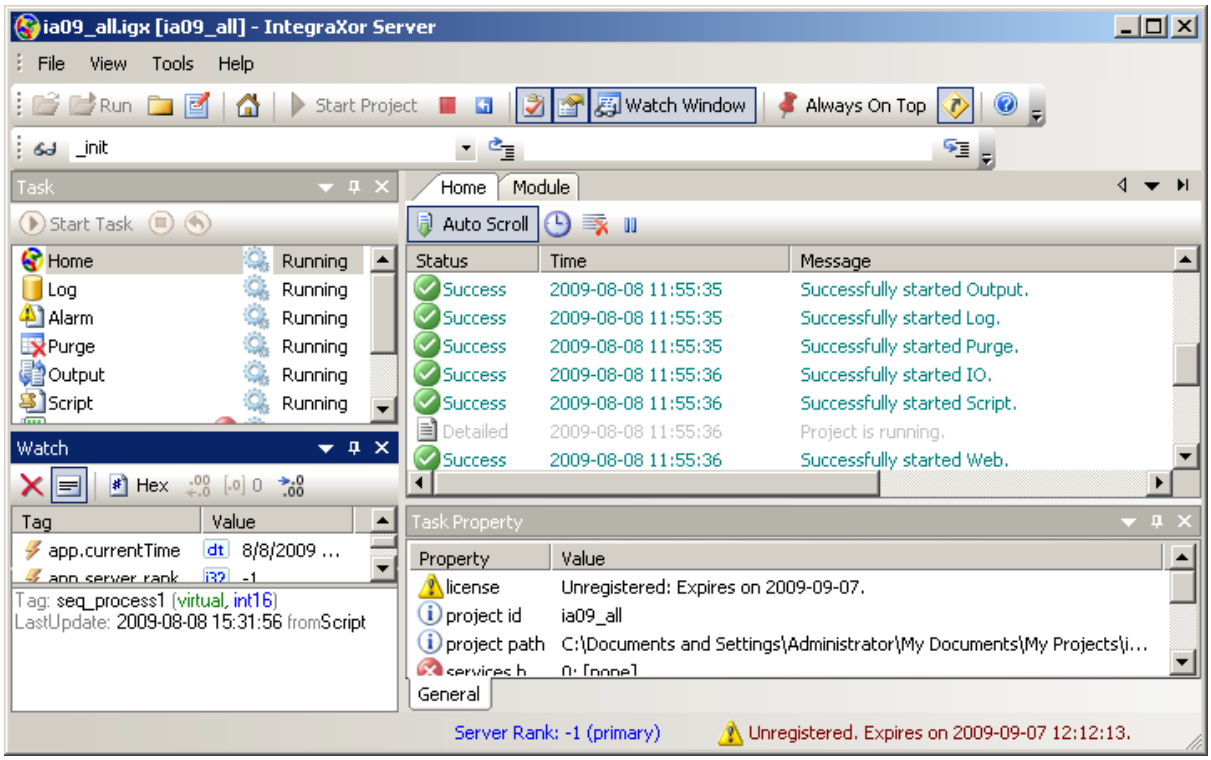
10. *Apply gradient effect*

    Shortcut key is 'g'.

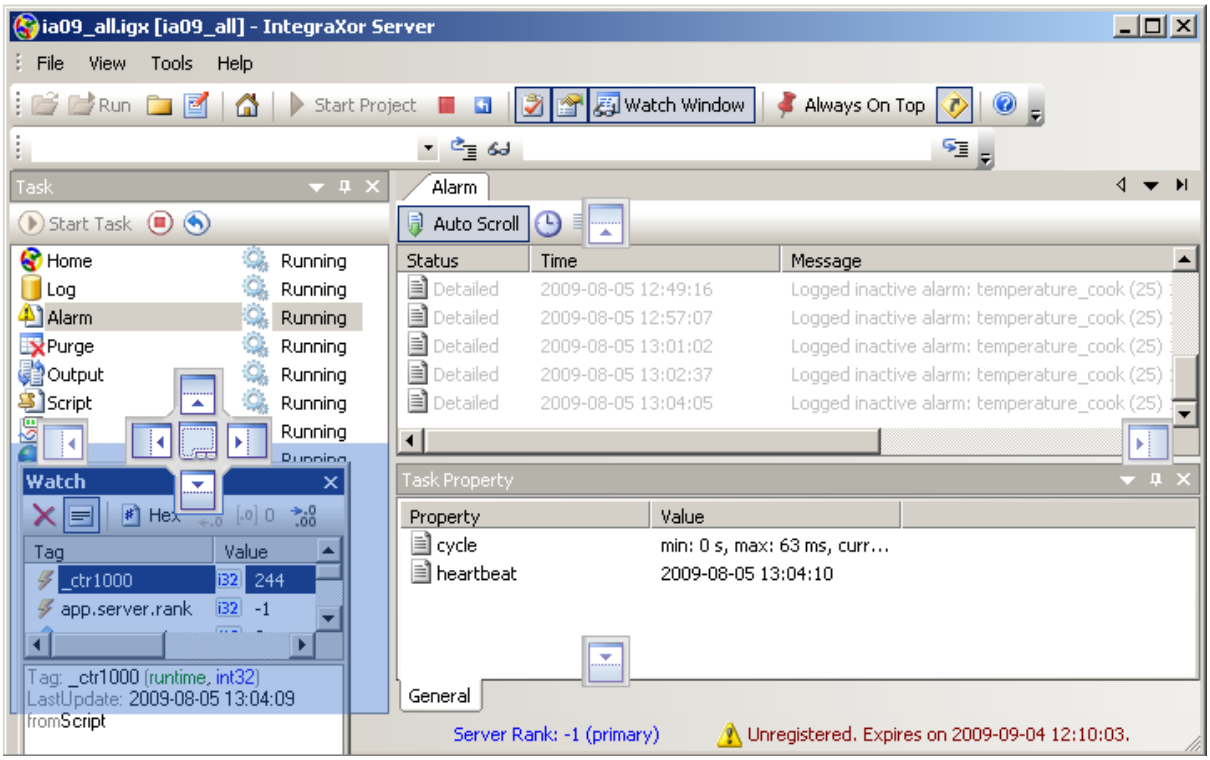
More information and tips are available in another sub chapter Working with Inkscape

Chapter 3. Server User Interface

The IntegraXor server interface is pretty straight forward and self explained. The layout consists of four major panes and then title bar, menu and icon bar at the top. The first main window is Task Window (Ctrl+Shft+T) which allows user to select the interested task for viewing in Status Window which normally showing Home on its window tab. Task Property Window (Ctrl+Shft+P) further display more information on the current condition of a selected task, for instance, overview of device status for I/O. Lastly, Watch Window (Ctrl+Shft+W) allows user to monitor a list of tags during development and troubleshooting.

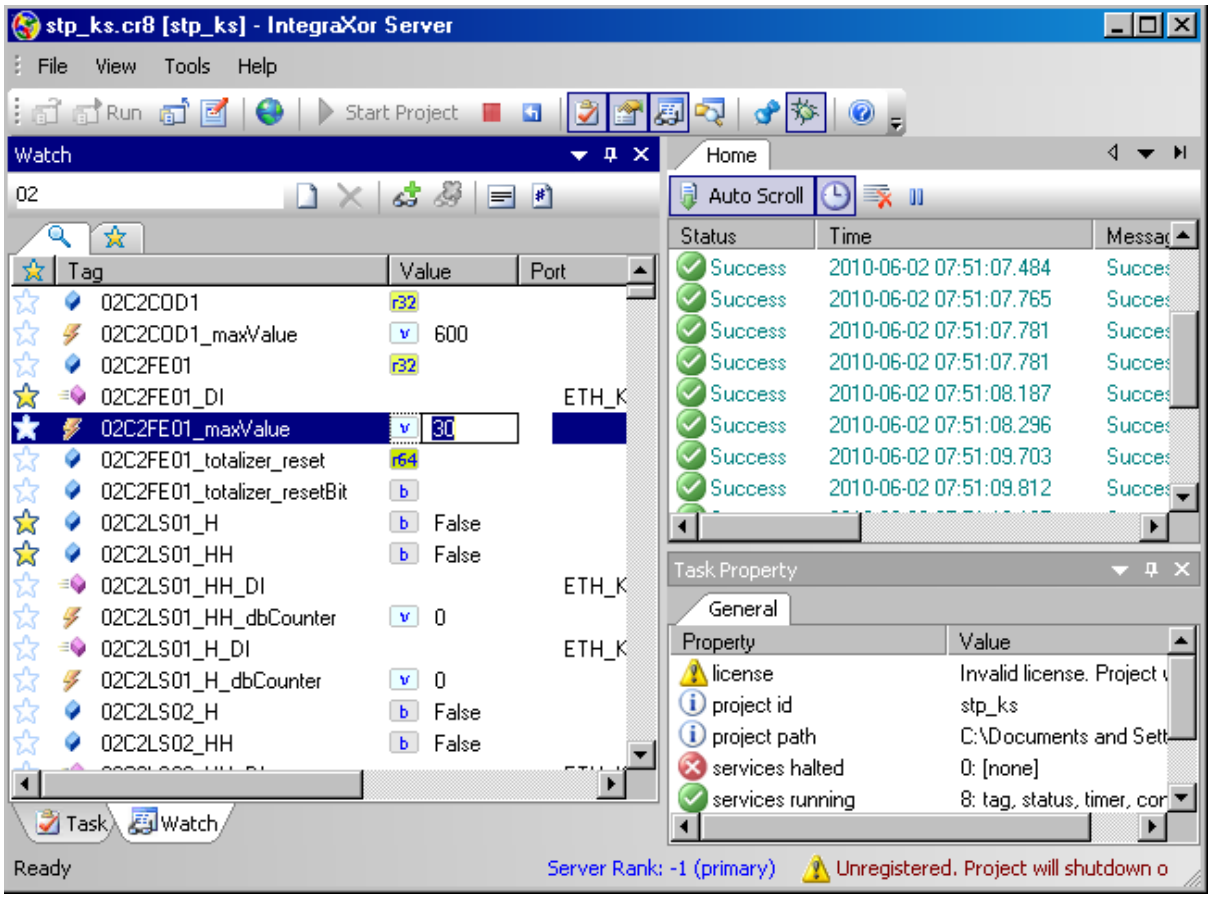


IntegraXor main interface.



Four panes can be dragged around, put to auto-hide or float for further customization.

The title bar always showing the project path. A project can be started or stopped online by using the Project Start/Stop button on the icon bar, moreover, the specific task can also be started by using the Task Start/Stop button on the icon bar to restart a specific task independently after a new configuration is entered. To do an online restart on a speific task, do select the desired task on the left pane, then the Task Restart button will be enabled.



Tag watch listing for configured tags.

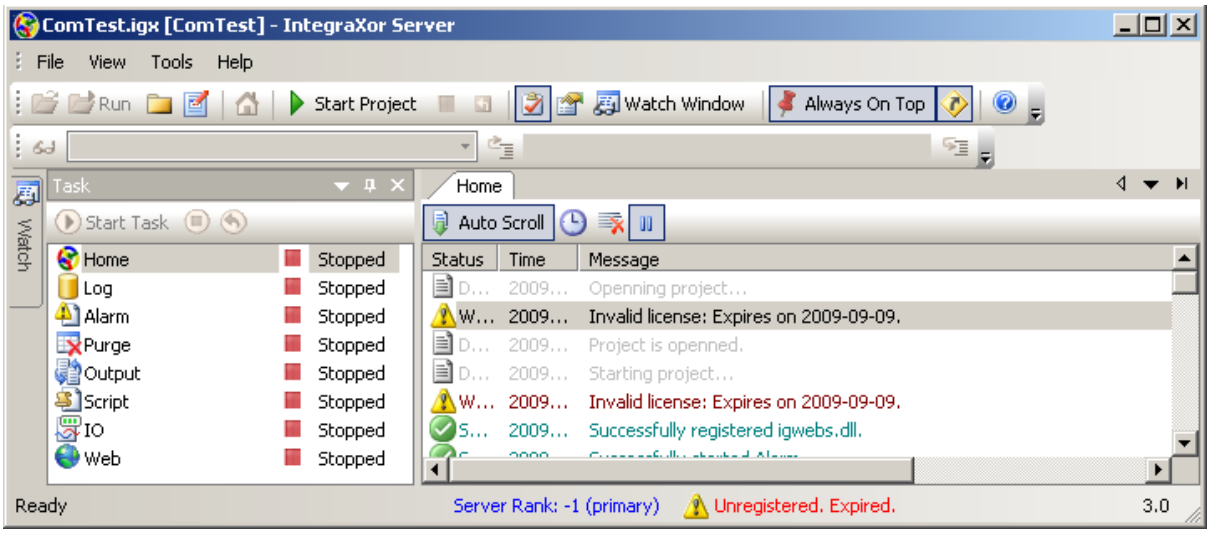
Tag values monitoring or manipulating for troubleshooting can be done via Tag Watch window. Use the filter on top left to narrow down the search, and click on the value to change tag value. Click on star to add to Favorite Tab, or create new Tab for customized tag list of monitoring.

Note

Although all tasks can be updated online by restarting the specific tasks, or even restarting the whole project, however, tags in the watch list won't be updated if there's any new tag added or old tag deleted. IntegraXor server must be totally exit and restart to refresh the watch list.

3.1. Licensing

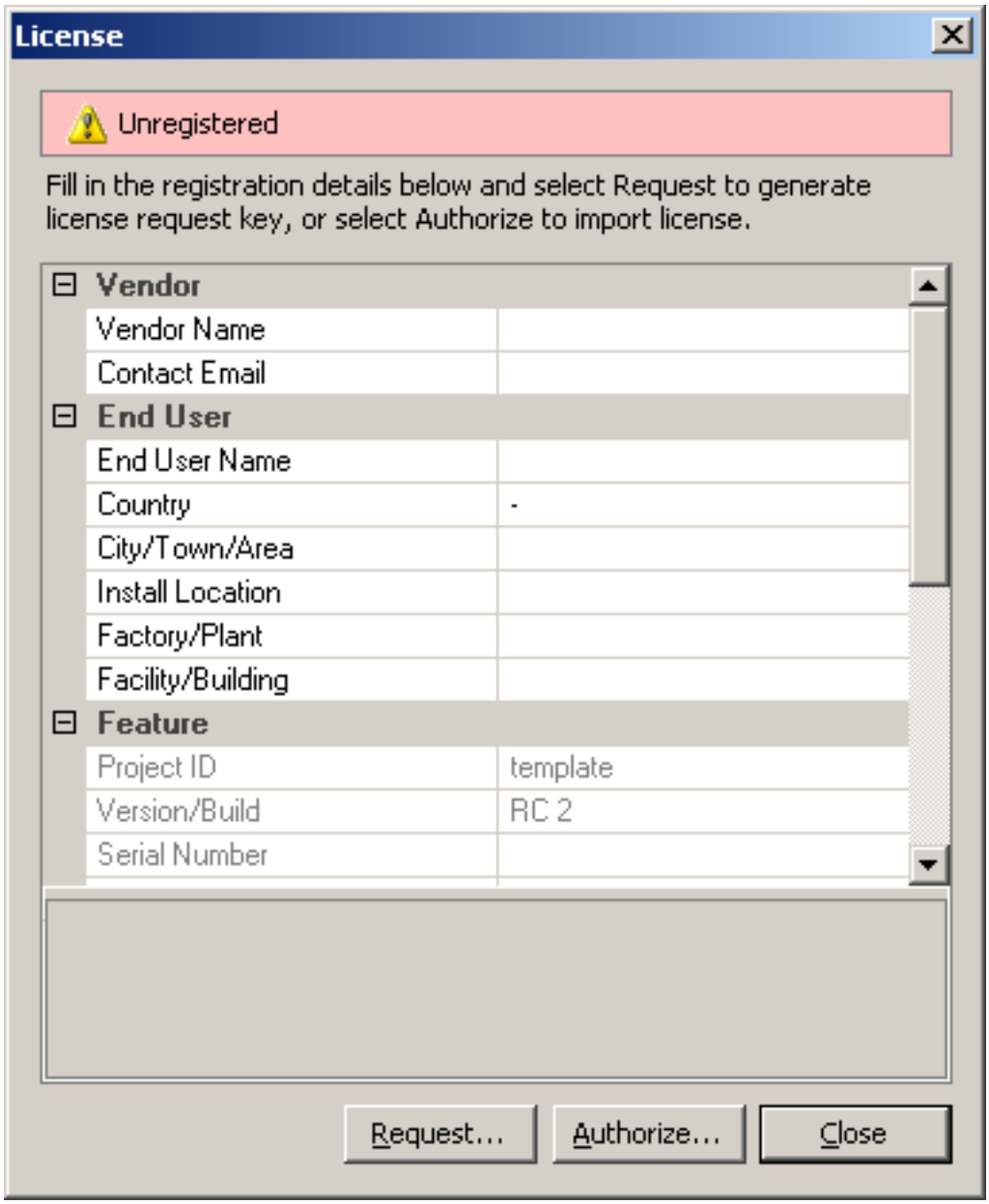
IntegraXor is a commercial industrial software. It runs with full features in demo mode by default which shall expire in 2 hours everytime it is started. A message will be prompted at Server's status bar upon expiry, just exit the server and start over again and no reinstallation is needed. Registered server will run permanently even after any promotion ends so long authorization is performed successfully.



Note the status bar showing license is expired at the bottom.

The process of registration is such that the user shall generate the registration file `*.ilr` on the project designated computer, then send the registration file to your local sales agent or distributor, or directly to register@integraxor.com. The software vendor shall return another authorization file `*.ila` or software key based on the project and hardware configuration on that particular machine. User may request for another authorization key if migration of hardware is needed or upon encountering hardware failure.

Its status can be checked from the menu, Help, then "License...". A pop up like below shall show the current licensing status.



License status and user information.

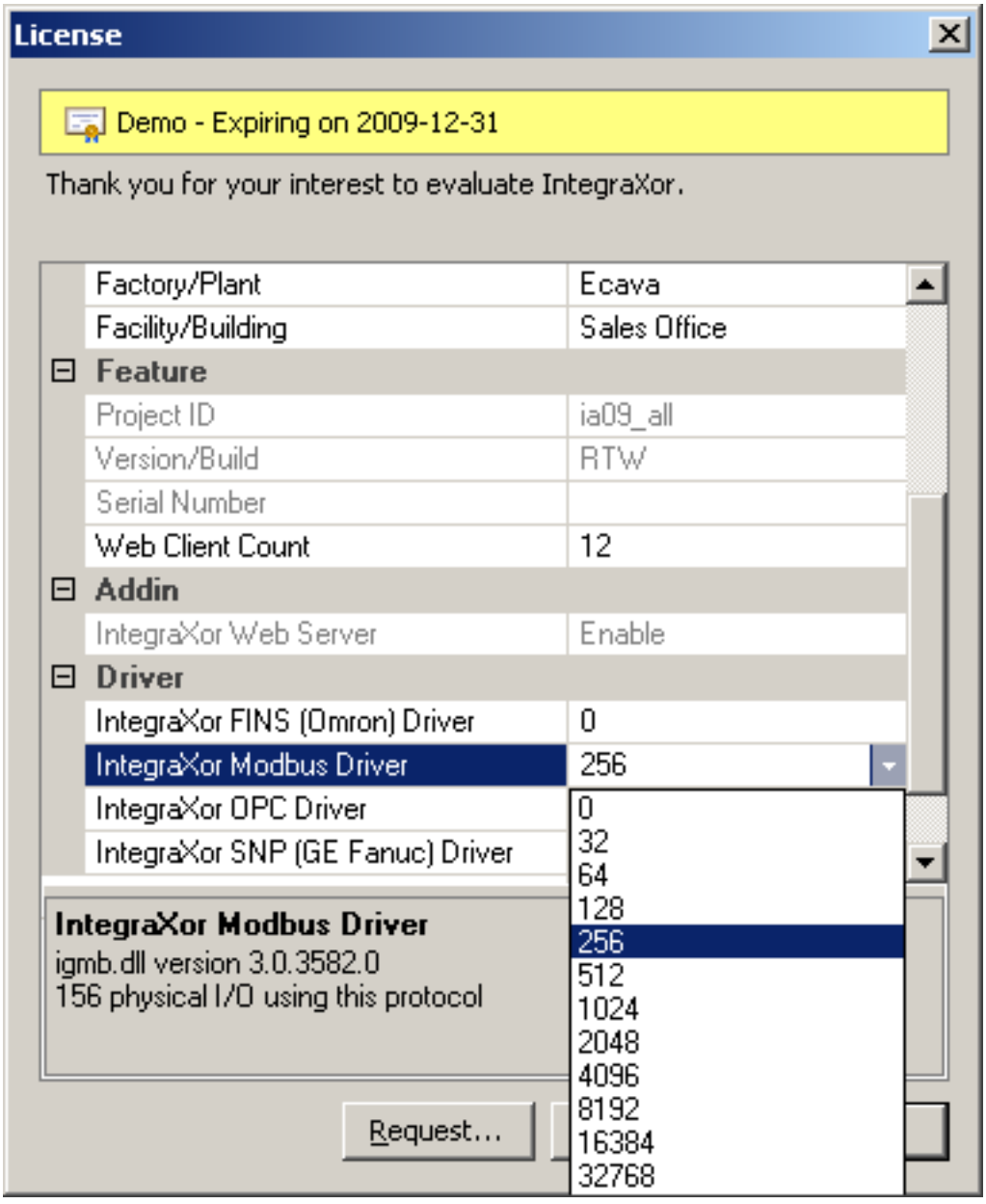
All details must be entered truthfully and correctly. Use mouse click or enter an arrow key to navigate from one field to another. The "Contact email" is especially crucial, the authorized file will be sent to the entered email address. Any future upgrade or changes will still based on this email address or at least another user name based on the similar web domain name, as such private web based email address shall be avoided. Once all fields are entered, the Registration file *.ilr can be generated by clicking the "Request..." button, and choose the desired directory.

It's highly recommended to keep all licensing files `*.ilr & *.ila` in the project folder itself. Note that the licensing is based on machine and project itself. As such user may purchase more licenses if multiple projects needs to be run on one single machine.

The Project I.D. is the main reference to the project that used in many places, especially the web address link, databases and so on. The default project I.D. 'template' and other generic names like 'project' or 'demo' which can be used repeatedly will be rejected. A good project ID should not contain spaces or odd characters to avoid any difficulty to type it out, as it will appear as part of the web address. The Web Client Count or Remote

Concurrent Connection is the desired number of *remote* mimic viewer, any connection from different IP regardless of LAN or WAN will be counted, but local viewer on the server for stand alone system is not counted.

The Physical I/O Count column allows user to select the number of I/O count or tag that associated with field devices' data required, regardless of the data type used. Note that the suggested I/O number is always slightly higher than the used I/O tags, re-select a higher number if more I/O tags is needed. The counted number of I/O used is shown on the property box when the relevant driver is selected. Note that virtual/memory/runtime tag count is excluded and no limitation is imposed.



Demo license. Also note the tag counted shown at the far bottom: "156 physical I/O using this protocol"

Once the Authorization file is received, simply open up the License Window again and choose to import authorization code by clicking on 'Authorize...'. Browse the authorization file and accept it. These licensing files are encouraged to be stored in the project directory itself to avoid misplaced.

In certain circumstances, demo license with longer/specific expiry date can be requested. It's useful for longer

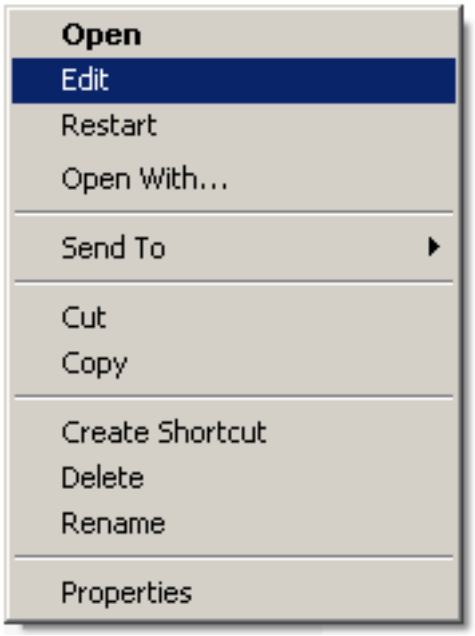
evaluation period, or longer development on a temporary PC before the designated site PC is available, or for delivering a project to site/end user, when the payment is still outstanding.

Important

Once the system is authorized or licensed, the registered information is *not* allowed to change.

Chapter 4. Back End Configuration

All back-end tasks shall be configured by IntegraXor Project Editor. Right click on *.igx file within the Windows Explorer and select 'Edit' to start the configuration work, Project Editor will be launched automatically.



Windows Explorer shell integration

4.1. General Information/Convention

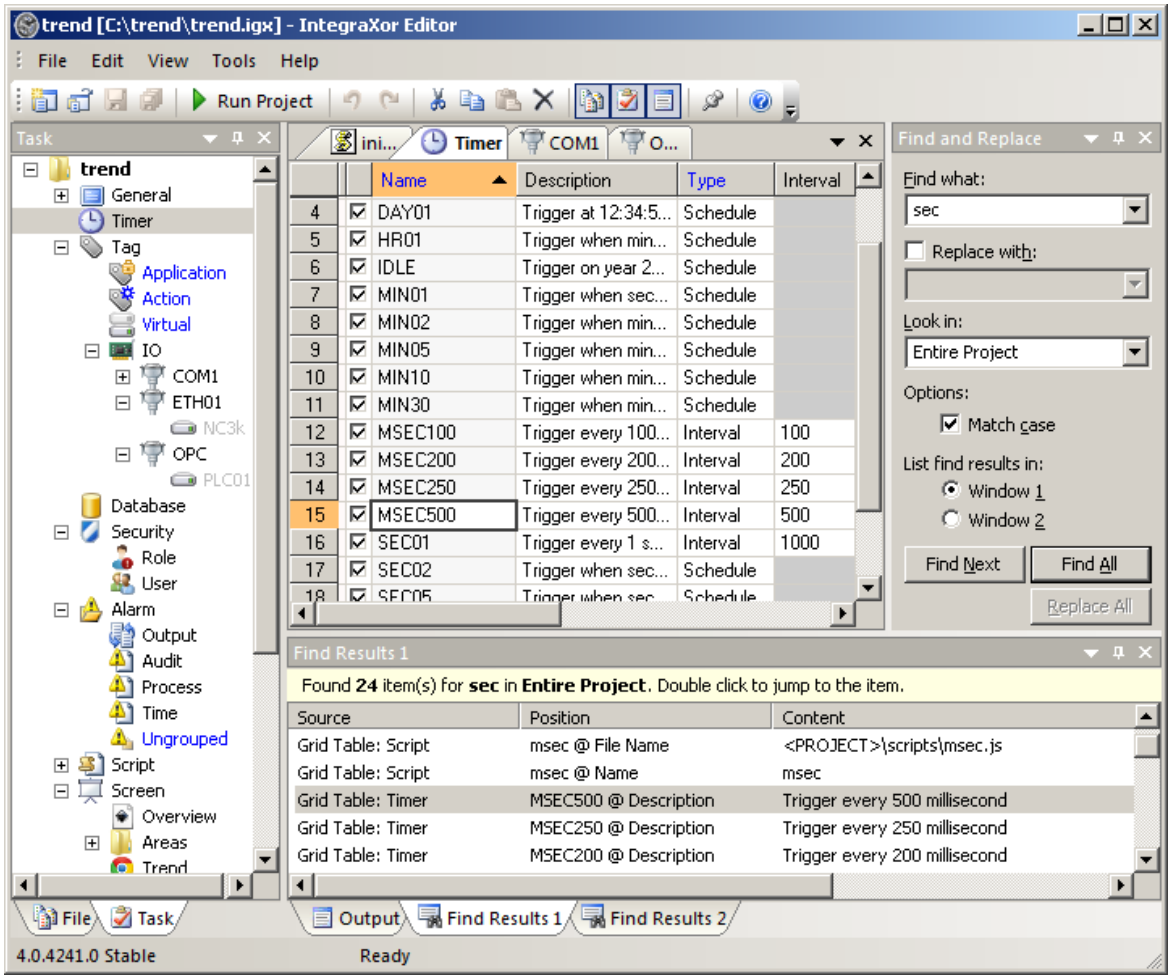
- All changes must be followed by a task online restart or complete project online restart so that it could take into effect.
- Most tables contains of four essential columns, i.e. Name, Description & Enable checkbox. There might be different sets of additional column(s) depending on the task.
- Bold field indicate that the particular field or setting is mandatory, otherwise it is optional.
- *Italic* text indicates that the particular field or setting is meant for system use only, no intervention required.
- Remarks could be placed right after two slashes, //.
- The settings shown in this document is merely a working example.

Every task entry has a Name column which shall be entered with a unique name, it should contain only alpha-numeric character. Moreover, it should not start by any number or dot. Space, dash, mathematic & logical operators, and all other symbol shall be avoided.

4.1.1. Sequence of Configuration

Although it is not a must to enter configuration data in sequence, but it is best to practice such that the following sequence is observed.

- TIMER - Timer is the engine of most, if not all task. As such the Timer in the template is preconfigured with some standard periods of time, like SEC01, SEC02, MIN01, MIN05, HR01 and etc.



  Editing configuration.

- PORT - All data must have a mean or channel to communicate, and Port shall meant for the channel to communicate with the external devices. Devices and I/O tags are configured here.
- DATABASE - Database is needed for Trending, Alarm and some user data.
- ALARM - Alarm can only be configured after the relevant tag(s) is created.
- OUTPUT, SCRIPT, USER - These tasks do not need to be configured in sequence, unless Script may need some tags to exist to make it relevant.

4.2. General

General table defining all general project related information.

Parameter	Description
Project ID	Project ID defines the main reference of the project. It will be used at the web address link, databases and so on. It is advised to name a project ID short and precise just like ordinary web address, no space and no special characters are allowed.
Description	Additional description.

Parameter	Description
Title	Project title entered here will be displayed in the top-middle of the header.
Logo	Project logo image linked here will be displayed in the left of the header.

Post-launch	Any program entered here will be executed after the server is started. Normally the browser needs to be launched once server is started. The following entry will launch the system default browser with the project home page.
	`"<BROWSER>" "<HOMEPAGE>index.html"`
	Some browsers support kiosk mode or running in full screen. For instance Internet Explorer can run in kiosk mode with following command line:
	`"C:\Program Files\Internet Explorer\iexplore.exe" -k "<HOMEPAGE>index.ht`
	Example below shown a portable version of Firefox to be started instead. The command line syntax is dependent on the selected browser and could be obtained from its relevant web site.
	`"D:\PortApps\PortableApps\FirefoxPortable\FirefoxPortable.exe" "<HOMEPAG`

Pre-launch	Any program entered here will be executed before the server is initialized.

Tip

If more than one program need to be started, multiple command lines can be entered accordingly.

4.3. Timer

Timer can be described as the engine of the system, many task cannot run independently without Timer as trigger. There are two types of timer, one is 'Interval' which is fairly straightforward and the other type is 'Schedule'. The concept of entry for Schedule Timer is similar to cronjobs used in Unix/Linux environment, which is used to schedule commands to be executed periodically. Enter desired time of event into each time element. One or more figure can be entered into one column by using comma as separator. For instance, enter "0" into Second column will make a timer that trigger in every minute at 0th second. Another example is entering "0,30" into Second column will make a timer that trigger in every 30 second at both 0th and 30th second.

Parameter	Description
Name	Any unique name
Description	Additional description to the entry
Type	Interval/Schedule
Interval	Numbers in millisecond
Year	Four digits entry
Month	1~12
Day	1~31
Day of Week	0~6
Hour	0~23
Minute	0~59

Parameter	Description
Second	0~59

Tip

Regular used timers will be added by default upon creating new project.

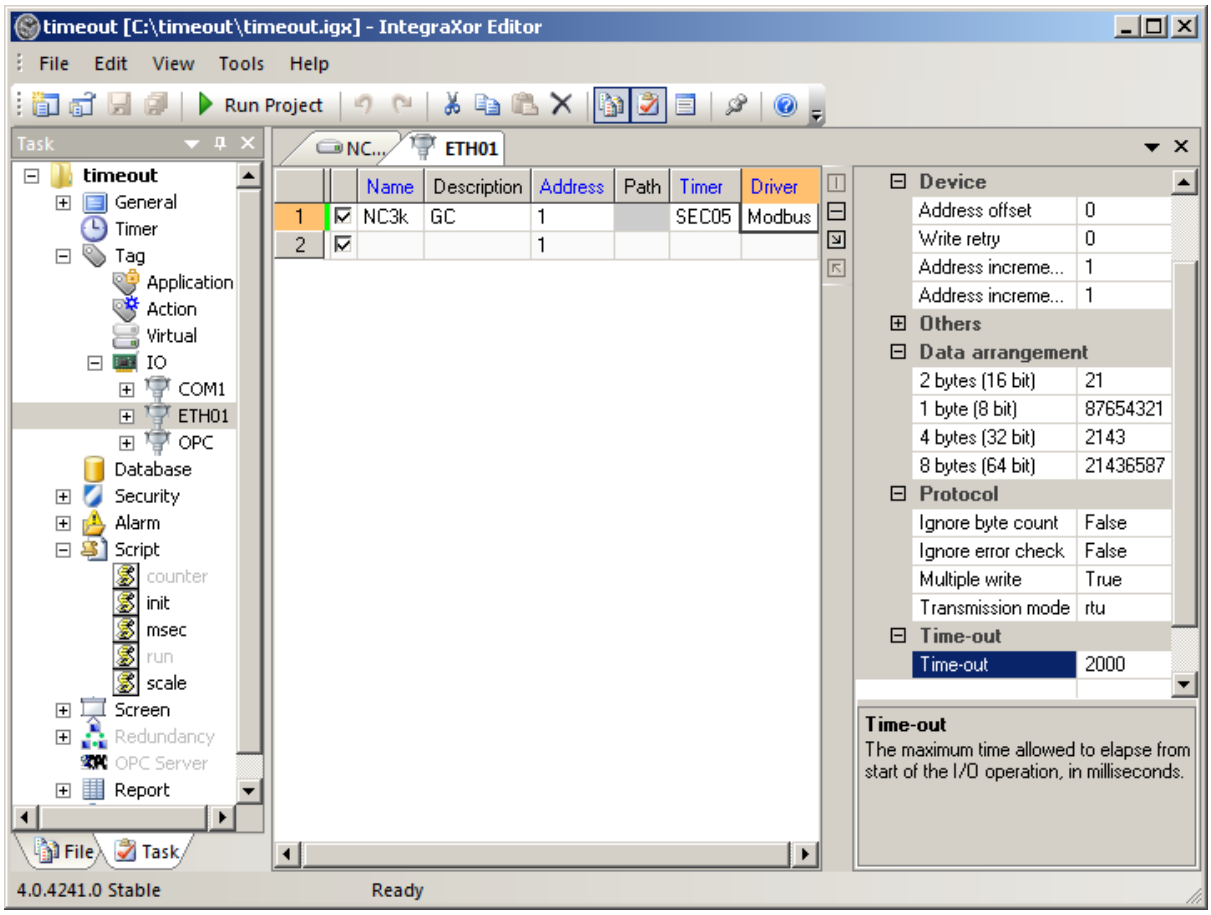
4.4. IO /Port

A system will not be called as HMI/SCADA if it is not communicate with any external device. Port is the first channel to determine in accessing the external devices. A Port can be a serial communication port, or it can be an ethernet port, or any physical connectivity to the external devices, for instance a proprietary PCI device interface card.

Parameter	Description
Name	A name to be referred from Device table.
description	Additional description.
Address	Physical port address. For instance, to specify serial port 1, type "COM1". And for Ethernet card, an I.P. address need to be entered, if port number is needed, it shall be separated by a column, e.g. "192.168.1.100:502" for Modbus TCP/IP server, or "localhost" for OPC server.

4.5. Device

One or more devices can be connected to the HMI/SCADA system via one defined Port. Device table will define the setting/behaviour of the connected device(s). A device can be a PLC, a RTU unit, a transmitter that possess communication capability.



Editing device.

Devices inherit from Ports, so it must be created under a desired Port.

Parameter	**Description**
Name	Any unique name for the configured device.
Description	Additional description.
Address	Numerical type node address, or device I.D. E.g. Modbus node.
Path	String type node address, or device I.D. E.g. OPC Server name: 'RSLinx OPC Server'.
Timer	Choose device polling rate.
Driver	This column specifies the protocol used.
Properties	Click on individual entry to see the details at the bottom.

Very often engineer needs to provide a page of network layout to display its connection status. Each device status can be obtained via system tags based on the name given by engineer. For instance, if a device is named as "PLC01", then a list of status tags will be available for relevant usage. The name of tags can also be obtained from Tag window in Server Interface. Engineer could click the dropdown list to check the available tags promptly.

Tag	**Type**	**Description**
app.device[PLC01].status	Integer	0 - Connected, 1 - Initializing, 2 - Disconnecting, 3 - Disconnecte Read Error, 5 - Write Error, Others - Unknown Status.

Tag	Type	Description
app.device[PLC01].status.connected	Bool	True/False.
app.device[PLC01].status.speed	Integer	Device responsiveness (ms).
app.device[PLC01].statustext	String	Connected / Initializing / Disconnecting / Disconnected / Read Error / Write Error / Unknown Status.

Tip

More than one device setting can be configured for a single physical device. For instance, one device may have different byte ordering for Float data as compare to Long Integer, then two device settings are needed.

4.6. Tag

Tag is a common terminology in general industry for variable in programming term. It was also called as Point in certain HMI system. Common SCADA system has two types of tag, one is IO tag and one non-IO tag. IO tag is tag that associate with external device data address, non-IO tag aka runtime or virtual tag or simply memory tag.

Tags inherit from Devices. To create a new tag, select the device and create an entry. The data type must be specified as per requirement. There are several types of accuracies for both integer and floating points in various vendor implementations. int8, int16, int32 are all integers, engineer must ensure what's the integer length of the type of PLC that being used, most common integer is int16. Same goes for floating point in industry devices, real32 is the most common floating point as compare to real64.

Parameter	Description
Name	Any unique name start by letter or underscore, avoid any other symbol like dash, dot, or special character or unicode. Try to make this a valid JavaScript variable name to exploit the power of Tag Expression. To be on the safe side for tag to work across most versions of browsers and scenarios, use only 26 English upper or lower case character and numbers.
Description	Additional description.
Data Type	int16/short int32/long real32/float real64/double string boolean int8/char uint8/BYTE uint16/WORD uint32/DWORD

Parameter	Description
Length	Length for string data type.
Batch	Batch numbering indicating if a chunk of data for one device shall be read or write together if they are contiguous in data address arrangement. It can be any number that can link them into one chunk or one batch, for instance, entering 100 for several contiguous tags will make them poll or update at one go. This shall speed up the communication a lot especially for serial communication. Modbus user can ignore this column but need to enter dummy tag to ensure the addresses are in one range, otherwise they will be polled separately. Note that these unused tags will still be considered into license I/O count, so engineer must arrange the addresses in contiguous form. Otherwise if changing the data address is not possible, then engineer need to optimize the license count and the need of combining several chunks of addresses.
Address	Data address for numerical type. E.g. Modbus Data Address. Not used for modern protocol which uses extended address like OPC.
Path	Data address for string type. E.g. OPC & GE Fanuc SNP. On the other hand, it's not used for protocol like Modbus.
Log	ID defined in Database Task. This shall automatically log the tag value once valid I.D. is entered. Which mean, this is the only thing need to do at back-end configuration if trending for a tag is required.
Read Level	*This column determines the tag's security level for reading. If the user login with user level lower than the value here, server will ignore the user's request to return the data. The highest security level is 1000 and it is set to 0 by default. It must not left blank.*
Write Level	*This column determines the tag's security level for writing. If the user login with user level lower than the value here, server will ignore the user's request to set the data. The highest security level is 1000, it is set to 0 by default. It must not left blank.*
Read Label	*Read Label granting the reading access for users who are tagged with the same Label, it was sometime intepreted as Security Role.*
Write Label	*Write Label granting the writing access for users who are tagged with the same Label, it was sometime intepreted as Security Role.*
Input Expression	*Input Expression provides a quick mechanism for user to perform data scaling, PLC analogue value to floating point conversion, and any engineering unit conversion. Entering a number will be treated as scaling factor like divided by a thousand can be entered as* `0.001`*. For more complicated conversion, like °C to °F, user may enter* `9 / 5 * $ + 32`*, whereby $ representing the raw value.* *User can even link several tag inputs into conversion. For a flow calculation that taking frequency as input, user may enter* `($ / $KFac) * 3600 * $MFac`*, whereby* `MFac` *&* `KFac` *are tags for Meter Factor & K-Factor input respectively.*
Output Expression	*Output Expression provides a quick mechanism for reversed data conversion, for instance, sending data out to devices. Entering a number will be treated as scaling factor like multiplied by a thousand can be entered as* `1000`*. For more complicated conversion, like °F to °C , user may enter* `5 / 9 * ($ - 32)`*, whereby $ representing the raw value.* User may also clip the output within High & Low limits by entering `Math.max( LIM_LO, Math.min( LIM_HI, $))`; whereby `LIM_LO` & `LIM_HI` clipped the desired output range.

Parameter	Description
Default	Default initialized value or fallback value upon communication error. However, if Retentive (persistence) value was set and a valid previous data was stored, then this default value will be ignored.
Retentive	Database I.D. defined earlier. This shall automatically save the tag value upon changes. This retentive feature comes handy when a tag needs to start up with previously set, used or received data. System will store the last good value so that this last good value will be used upon next system start up, or upon communication error. Retentive value will always overwrite Default value when presents.

Attributes of the tag can be modified at the property window on the right when a tag is highlighted.

Parameter	Description
Input Tag	Set to true for Read-Only tag. In Modbus Protocol, 1x coil & 3x register need to have this attribute turn ON. This mean, to get Input Coil, you need to create a Boolean tag with this attribute ON, to get Input Register, you need to create a Integer/Float tag with this attribute ON. Otherwise, this attribute shall remain OFF. Note that the value for Input Tag cannot be changed via any mean except I/O input.
Write only tag	If set to true, this tag will be flag as a write only tag and server will not read this tag from device.

Tip

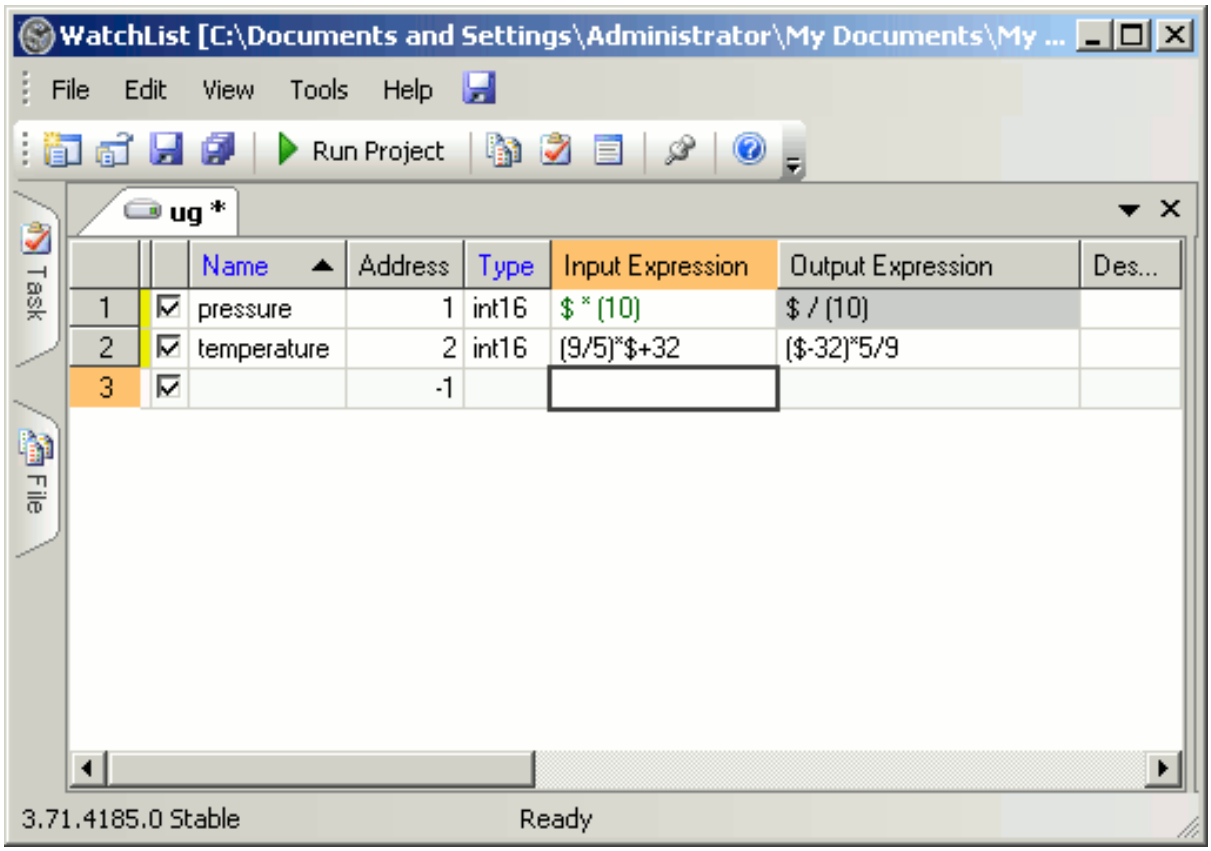
When a tag is renamed, all other tables which referring to this tag will be automatically renamed.

Note

1. IO tag (i.e. any tag that associated to physical device) cannot be set with value if the device is not connected.
2. Avoid to create any tag started by underscore '_' which might get conflict with system libraries' tags.

4.6.1. Tag Expression

For ordinary scaling requirement, user only need to enter any numerical number as needed. For instance, divided by a thousand can be entered as `0.001`, or scaling factor like multiplied by a thousand can be entered as `1000`. User will soon realize the entered number will be automatically added with some syntax with $ symbol when the cursor is away from the field. As per the examples above, entering `0.001` will change to `$*(0.001)`, and entering `1000` will change to `$*(1000)`. Take note that when the cursor is back to the expression column, the originally entered number will return and ready for update.



Expression for scaling/calculation

Tip

The source value can be omitted by adding `$*0+` in front of the code. And the auto scaling feature can be removed by entering `$+0`.

User may do any complicated engineering unit conversion like °C to °F, user may enter `(9/5)*$+32` as Input Expression, and `($-32)*5/9` as Output Expression. User can even link several tag inputs into conversion. For a flow calculation that taking frequency as input, user may enter `($/$KFac)*3600*$MFac`, whereby `MFac` & `KFac` are tags for Meter Factor & K-Factor input respectively.

Another practical PLC raw value linear scaling example, if raw value of `819` represent `10°C`, and `4095` represent `60°C`, then the input expression shall be entered as `((60-10)*($-819)/(4095-819))+10`, conversely the output expression can be entered as `(($-10)*(4095-819)/(60-10))+819`.

User may also adding fool-proof interlock by clipping the output within High & Low limits by entering `Math.max( LIM_LO, Math.min( LIM_HI, $))` into Output Expression; whereby `LIM_LO` & `LIM_HI` clipped the desired output range. For instance, entering `Math.max(0,Math.min(100,$))` will prevent operator from sending anything less than zero or bigger than hundred to the external device. For the sake of explanation, this Expression has the similar effect of entering `($>LIM_HI?LIM_HI:$)<LIM_LO?LIM_LO:$`.

It's also very common to perform bitwise operation [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators] with external devices which Expression can do the magic. To read LSB and MSB in one 16 bit integer, the expression can be entered as `$&1` and `$&65536` respectively (or reversely). On the other hand, to send bits out to an integer, Input Expression can be written as `$0+$b0*1+$b1*2+$b2*4+$b3*8+$b4*16+$b5*32+$b6*64+$b7*128+$b8*256+$b9*512+$bA*1024+$bB`

`+$bF*32768` whereby `b0...b9 & bA...bF` are the 16 boolean tags representing individual bit. System would expect some formular to be entered in Output Expression, simply enter single `$` symbol if no further manipulation of the value needed.

Important

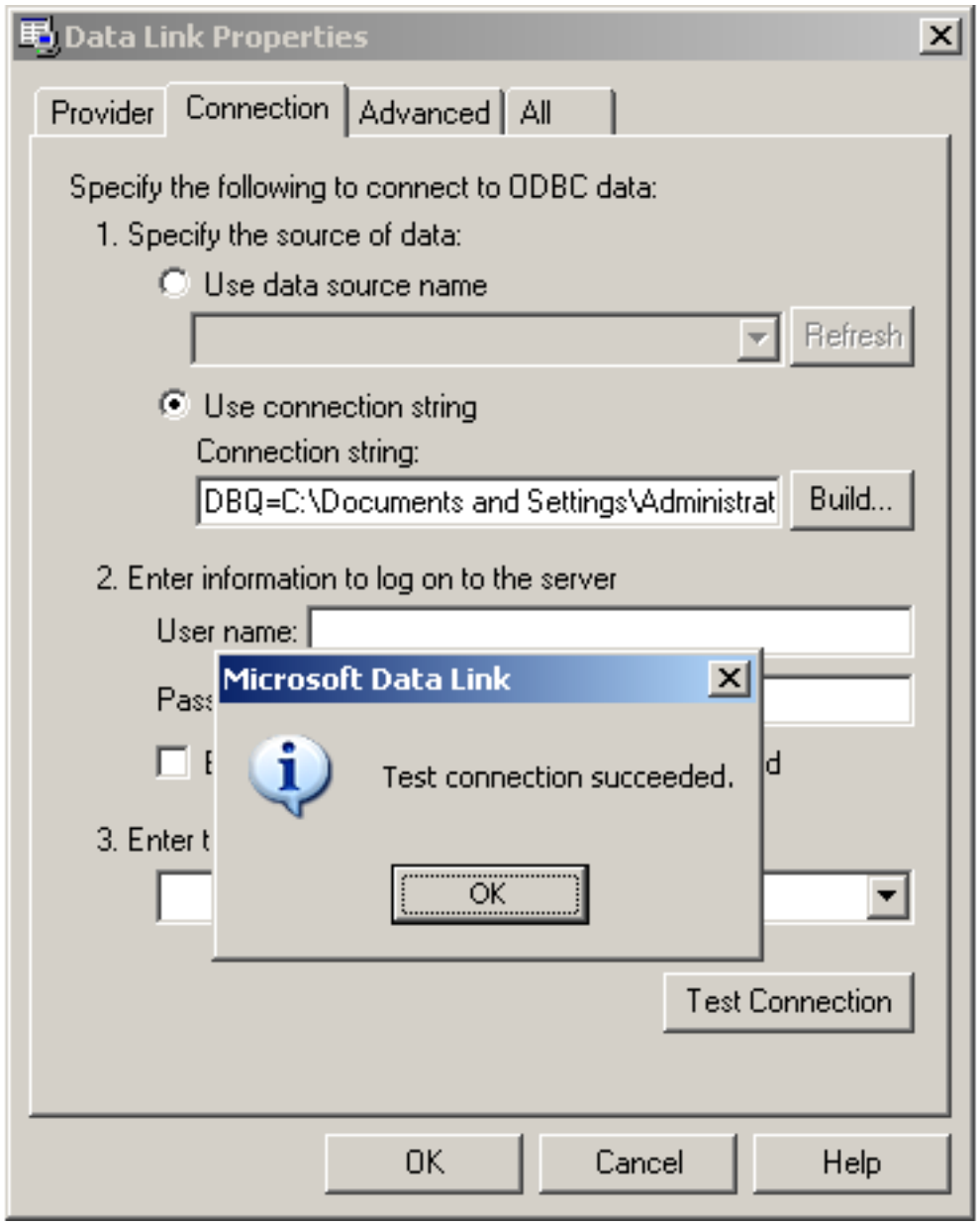
JavaScript doesn't perform bitwise operation beyond 32 bit. Using int64 or uint64 in bitwise operation will not produce expected output.

4.7. Database

Database is where an external database system is defined. It can be any ODBC supported SQL database. For instance Microsoft SQL 2005 Express edition, which is free for 4 GB of data storage.

Parameter	**Description**
Name	Any unique name to be referred in Tag or Alarm table.
Description	Additional description.

Connection String	This is a working example for Microsoft SQL 2005 express edition. In this database connection string, note that the DATABASE parameter need to be modified according to the database created earlier using database creation wizard. For instance if a database called my_job, meaning the parameter need to be written as "DATABASE=my_job". Note that for readability reason, the sample was split into lines, in actual scenario, enter the string in one line only.
	
	`Provider=MSDASQL.1;`
	`Extended Properties="DRIVER=SQL Native Client;`
	`SERVER=<COMPUTER_NAME>SQLEXPRESS;`
	`DATABASE=my_job;`
	`trusted_connection=yes";`

Parameter	Description
	 Clicking browse button [...] shall launch the Data Link Properties which could Test Connection on the string entered. More connection strings for other database can be found on this website: http://www.connectionstrings.com/. Do ensure the ODBC driver for the selected database is installed. For instance, MySQL ODBC driver can be downloaded from its provider at http://www.mysql.com/downloads/connector/odbc/ **Note** Always use only 32 bit ODBC driver for any brand of database even if the Operating Systems is 64bit.
Log	Define how frequent a data will be logged.
Purge	Define how frequent the server will perform purging sequence.

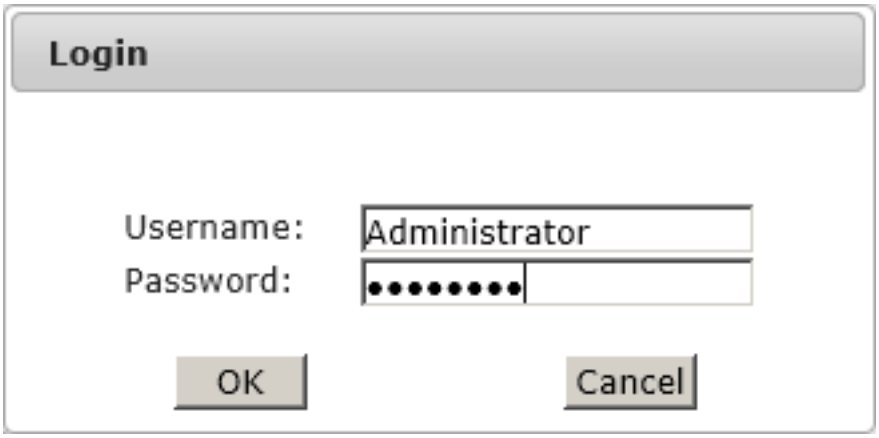
Parameter	Description
	Warning Purged database is permanently deleted.
Purge Count	Define how long or how big a database shall be kept. For instance, a value of 90 means data will be stored for 90 days only. Based on FIFO logic, the 91st day old data will be deleted during purging process.
Purge Type	Always fixed to 'day count' only. System works on First In First Out basis, i.e. old data will be deleted first.

4.8. User

User is an Account Management section where engineer could add user and its previlege (access level) into project. This security level checking is applicable on front end only, i.e., when request is submitted via http request from client side scripts. Security level checking is not applicable from within tag window of IntegraXor server, as server side execution gains the highest execution rights.

When a user security level is equal or higher than tag security_write level, he or she has the right to change the value of this tag. Be aware that default configuration put no security control to all access, that is security level of zero (0). Meaning all user can do anything without logging in. To impose security check, engineer must first increase security level for the relevant tag(s).

User login is per login per IP basis. This means same user can login via different IP address **concurrently.** System only maintain single login session per IP. This means only one user can login from one IP at any time. In order to login as another user, the first user must logout.



User login popup.

When user is attempting to login, browser shall prompt user a standard login dialog. User shall then key in user name (case sensitive) and password (case sensitive) and press OK to login. If the supplied user name or password is incorrect, browser will prompt the login window for another two times. User need to click on login button again to call out login window if fail to login for three attempts continuously.

Parameter	Description
Name	A case sensitive user name.
Description	Additional description.
Password	A case sensitive password.

Parameter	Description
Expire	Expiry date in DD-MM-YYYY format. Enter 01-01-9999 in year for non-expiry user.
Maximum Instances	Security feature that defined user maximum login instances or session at the same time in multiple locations.

Important

Expiry column must not left blank, otherwise user will not be able to login.

4.8.1. Web Access Information

To login, web client should issue a login URL in the following format:

`http://ip_address:7131/PROJECT_ID/login?`, where *`PROJECT_ID`* shall be changed according to the project name given.

S.I. engineer needs to check the returned HTTP header for IgUser tag for login status. If IgUser exist, login was successful. The IgUser format is as follow:

`IgUser: {name:'xxxx', level:xxx}`

S.I. engineer can then use JavaScript `eval()` function to convert the string to a valid JavaScript object to access the current login username.

Once a login is granted, browser will maintain the security credential and will use the same login info to contact IntegraXor server for all subsequent request without user explicit request. Browser will only drop this behaviour when user close the browser window.

In order to logout explicitly, web client should issue a logout URL in the following format:

`http://ip_address:7131/PROJECT_ID/logout?`, where *`PROJECT_ID`* shall be changed according to the project name given.

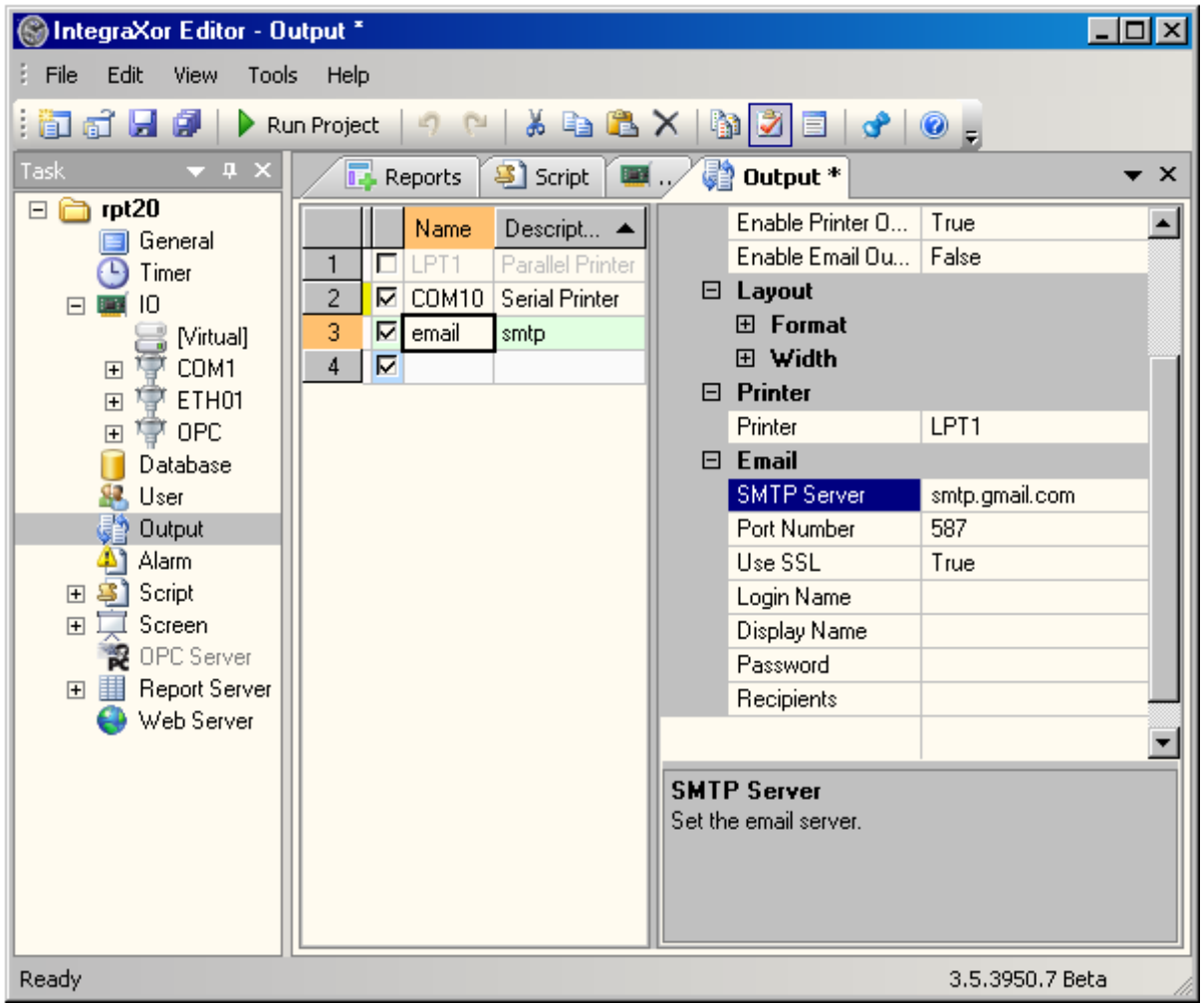
This shall instruct server to untie this IP from any username. But in HTTP header traffic, browser still keeps sending the previous credential to IntegraXor server, however, server will ignore the credential since it is not registered in server security login list anymore.

4.9. Alarm

Alarm is where engineer could configure a list of alarm or event that required by project. Alarm can be treated as Event if needed, engineer only need to group them with another different name.

Parameter	Description
Name	Name of the alarm, which can be associated to scripting control.
message	Message string to be display or log into database.
group name	Group naming. Example for Alarm Type is System, Hardware, Operation. It can also sectionize into area or priority.
log to	This is the link to Database which indicate which database this alarm shall store.
output to	This is the link where the alarm string shall send to. It can be configured to send to serial (COM) or parallel (LPT) printer, or send out as SMS/email alert.

Parameter	Description
tag name	This is tag entered in Tag table. An alarm shall behave mainly based on this tag.
triggered by	There are three triggering type: compare value compare tag on change.
condition	Comparison conditions: ==: equal <>: not equal <: less than >: greater than <=: less than or equal >=: greater than or equal
limit (compare item)	This is the limit which can be a tag or value depending on comparing type.
deadband type	Treat deadband parameter as value, percentage or time delay in second.
deadband value	Dead band for comparison.
type	Either make this an ordinary alarm or audit tracking only. Alarm Audit
Description	Additional description.
name	Name of alarm.



Output Configuration to Printer, SMS or Email

Tip

If there are several tags that needs to be combined into one tag, a sample script like below can be of use:

```
var S1 = getTag ( 'SIM1_Lek');
var S2 = getTag ( 'SIM1_Breuk');
var S3 = getTag ( 'SIM1_Service');
var S4 = getTag ( 'SIM1_Loopim');

//turning on anyone of the tag will trigger the event.
setTag( 'SIM1_Alarm', S1 | S2 | S3 | S4);
```

And this script can be further simplified into one line in a virtual tag's Tag Expression.

```
$SIM1_Lek | $SIM1_Breuk | $SIM1_Service | SIM1_Loopim
```

4.9.1. Additional Information for Deadband

Deadband is needed for a value that flicker or fluctuate a lot. If a tag value keep fluctuate around the alarm limit, than similar alarm may keep coming and going off so generates unnecessary repeatated alarms. This problem can be avoided when deadband is introduced, so alarm shall consider go off only when the value go below limit

plus deadband ether in actual value or in percentage.

For instance, if high limit is set to 80, and deadband set to 10, then alarm shall occur when value greater than 80. However, alarm will not go off when value fall across 80, but alarm shall consider off only when value fall below 70. Obviously alarm shall reoccur as usual if it rise above 80 again.

Deadband also can be entered as time delay in seconds, whereby alarm will be considered as once if it oscillates multiple times across the limit within the specific time frame. On the other hand, alarm will *not* be triggered if it doesn't stay longer than the specified duration. For instance, a spike that runs less than entered duration will be ignored.

4.9.2. Alarm Scripting

IntegraXor will automatically generate two system tags based on entered alarm name. First is the enabling control tag which will be in the form of `app.alarm[Group.Name].enable` and then the second is the status of alarm that construct in the form of `app.alarm[Group.Name].status`.

State	Value
Active & unacknowledged	4
Active & acknowledged	2
Inactive & unacknowledged	1
Idle	0

The individual status will be combined using OR bitwise operation into group status `app.alarm[Group].status`. This status tag can be used to in Trigger column of Script task for further action when particular alarm occurs.

4.10. Script

Most of the scripts (or logics) can be placed and execute within the browser or at the client side. However, there may be time that some logics doesn't need to be interact with user via web page. Or, some critical logics need to be running at all time regardless which page user is viewing and even when browser is closed. These scripts can be run from server side and they can perform most efficiently especially when scripts need to access server tag values.

On top of standard JavaScript function, IntegraXor comes with a few additional functions for server relevant execution. Two most commonly used functions are *`getTag('tagname')`* and *`setTag('tagname', data)`*.

Parameter	Description
Name	Script name to be called when 'trigger_type' is set as 'User'. For consistence sake, always make script filename same as this name.
Description	Description of the server script.
File Name	Directory to the Javascript file.
Triggered By	Method to trigger the script. Can be triggered by predefined timer, or change of value of a specific tag.
Trigger	Depending on the column "Triggered By", this column can either be a predefined timer in Timer section, or a tag in I/O section.
ActiveX	Enable or disable the ActiveX functions.

Parameter	Description
	Enable ActiveX only when need to use ActiveX functions to interact with Windows O.S.

Tip

Use system tag `app.startup` to trigger any script that only need to run once during startup, typically any script that needs initialation.

Use alarm system tag to trigger any script on any event.

4.10.1. Additional Functions

IntegraXor comes with a few additional proprietary functions or methods to simplify the programming work, mainly for script to communicate with the server.

setTag('TagName', Data) is used to set Tag value, where TagName is the tag in string, and Data is the variable holding value to set into the specific Tag.

getTag('TagName') is used to get Tag value, it returns the value in associated data type.

sleep(DELAY) is used to hold the script execution for specified time, DELAY in millisecond. setTimeout() and setInterval() may not work on server side in most conditions.

print('url', DELAY[, PRINTER]), this function allows a page (url) to be printed without invoking the standard printing dialog hence makes automatic printing possible. There may be scenario that certain data needs to be polled and execute which may take out some time, so engineer shall introduce some delays (the second parameter, DELAY) in millisecond to ensure the page is ready before printing. The third parameter PRINTER is an optional parameter which refering to printer name, default printer will be selected for printing if printer name is not supplied, or when the specified printer is not found. This function is needed for periodic printing, for instance, daily or hourly report. This print function always print to system default printer.

Tip

Using `getTag( 'same_tag')` to fetch data from same tag can introduce unnecessary overhead on CPU usage that may impact performance especially on large project or slow CPU. A better approach is to create an internal variable to cache the value then use the variable repeatedly.

```
var max = getTag( 'Tag001');
for (var i=0; i<max; i++) {
    //do something
}
```

4.10.2. Advanced Scripting

For advance user, the server side script can be called or initiated from client side script by using the following http request:

```
POST /run?script=script_name;param1=value1;param2=value2
```

A simpler approach to invoke server side scripting without passing in any parameter is to set the Trigger Type to 'Tag', then change the tag value from client side to invoke the said Server Side script.

setTag() does accept optional parameter whereby a virtual tag's time stamp can be altered for database logging purpose. `setTag( 'tag', data, option);` Where the `option` is a JSON object that has a `time` item accepting Javascript `Date()`. A complete sample can be seen in the following snippet.

```
var opt = {
    "time" : new Date(
            2020, //yyyy
            0, //M; 0: Jan, 1: Feb...
            1, //D
            1, //h
            2, //m
            3, //s
            456//msec
    )
};
setTag( "myTag", 123.45, opt);
```

4.10.3. Database Functions

getSql() provides convenient way to perform SQL query transaction which return result as `string`. Below is the usage and its parameters.

```
getSql([string] sql, [object] option)
```

```
[object] option format:
{
  dbgroup: 'xx',  // [string] database name created in database table,
                  // default to all database if not supplied.
  offset: 0,      // [integer] skip the first stated number of result, default to no offse
  limit: -1,      // [integer] limit to return stated number of result, default to no limi
  skip: 0,        // [integer] skip certain number of rows between results, default to no
  outfmt: 'xx'    // [string] output format, valid values are csv/csvh/json/xml, default t
}
```

Sample usage: `var result = getSql( "select * from log", {dbgroup:"dblog",offset:2,limit:3,skip:2,outfmt:"json"});`

Graphical representation of an example of the parameters usage. Note that only record 2, 5 and 8 are returned, the rest are ignored.

Record	Parameter
0	Offset 1
1	Offset 2
2	*Result 1*
3	Skip 1
4	Skip 2
5	*Result 2*
6	Skip 1
7	Skip 2
8	*Result 3*
9	Skip 1

10	Skip 2
11	Result 4
12	...
...	...

setSql() provides convenient way to perform SQL query transaction without returning anything. Below is the usage and its parameters.

```
setSql([string] sql, [object] option)

[object] option format:
{
  dbgroup: 'xx'    // [string] database name created in database table,
                   // default to all database if not supplied.
}
```

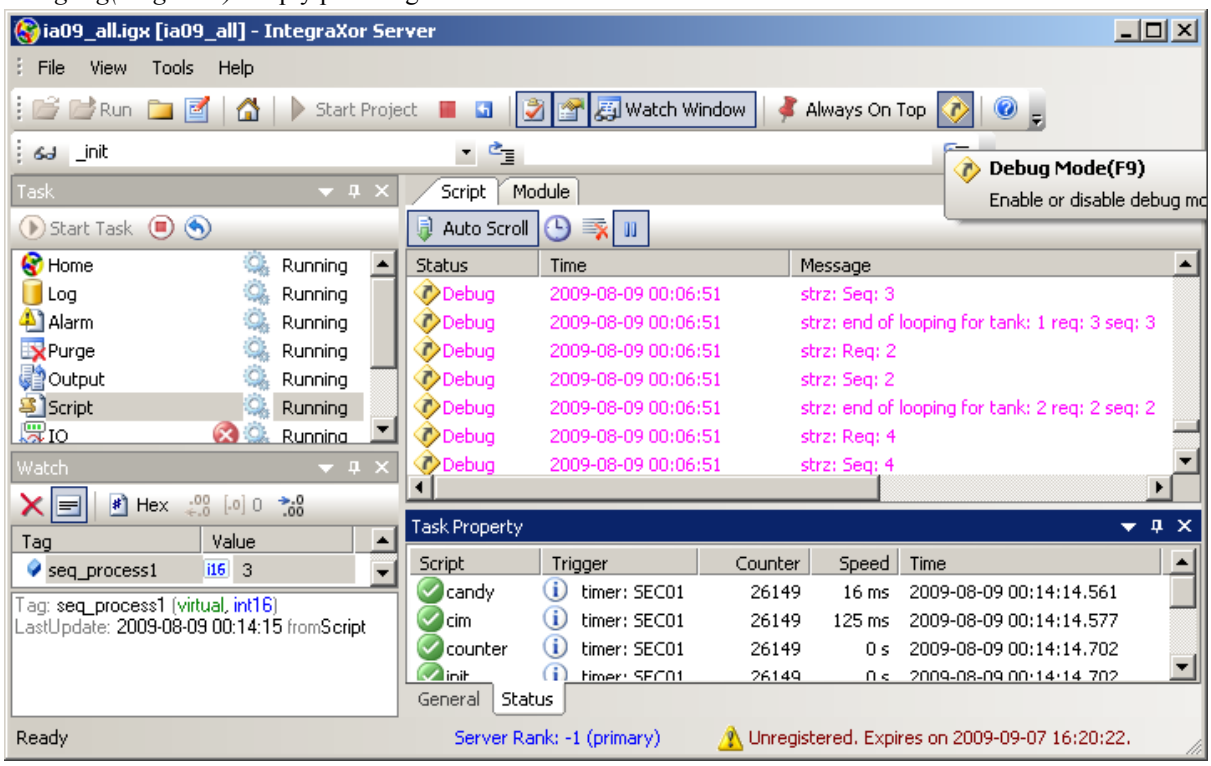
Sample usage: `setSql( "insert into alarm values (cell1, cell2...)", {dbgroup:"mdb"});`

4.10.4. Debugging Functions

More functions are available for server side only, they are mainly for debugging purpose. These functions required debug mode to be turned on by clicking on Debug button, the debugging information will be displayed on the Status Window when Script is selected on Task Window.

debugString(String) is used to send out debugging string into Status Window.

debugTag(Tagname) simply print Tagname and its value onto Status Window.


ia09_all.igx [ia09_all] - IntegraXor Server
File View Tools Help
Run Start Project Watch Window Always On Top
_init
Debug Mode(F9)
Enable or disable debug mo
Task
Start Task
Home Running
Log Running
Alarm Running
Purge Running
Output Running
Script Running
IO Running
Script Module
Auto Scroll
Status Time Message
Debug 2009-08-09 00:06:51 strz: Seq: 3
Debug 2009-08-09 00:06:51 strz: end of looping for tank: 1 req: 3 seq: 3
Debug 2009-08-09 00:06:51 strz: Req: 2
Debug 2009-08-09 00:06:51 strz: Seq: 2
Debug 2009-08-09 00:06:51 strz: end of looping for tank: 2 req: 2 seq: 2
Debug 2009-08-09 00:06:51 strz: Req: 4
Debug 2009-08-09 00:06:51 strz: Seq: 4
Watch
Hex
Tag Value
seq_process1 i16 3
Tag: seq_process1 (virtual, int16)
LastUpdate: 2009-08-09 00:14:15 fromScript
Task Property
Script Trigger Counter Speed Time
candy timer: SEC01 26149 16 ms 2009-08-09 00:14:14.561
cim timer: SEC01 26149 125 ms 2009-08-09 00:14:14.577
counter timer: SEC01 26149 0 s 2009-08-09 00:14:14.702
General Status
Ready
Server Rank: -1 (primary)
Unregistered. Expires on 2009-09-07 16:20:22.

Debugging information scrolling through in Pink text on Status Window.

Below is a sample code that demonstrates the use of debugging functions for troubleshooting. It can be copied and pasted into an script entry to see the program flow.

```
var a = getTag( "tagA") || 1; //default value 1
var b = getTag( "tagB") || 2; //default value 2
var c = getTag( "tagC"); //no default value

if (c > 1000) {
    a = 1;
    b = 2;
}

a = a + 1;
setTag( "tagA", a);

b = b + 2;
setTag( "tagB", b);

c = a + b;
setTag( "tagC", c);
debugTag( "tagC"); //this is a sample use of debugTag().

//debuging function will be executed in debug mode only
debugString( "Script is successfully executed."); //keep this as last line.
```

4.10.5. ActiveX Scripting

When ActiveX is enabled, the scripting engine will be based on Microsoft Windows Script Technologies [http://msdn.microsoft.com/en-us/library/d1et7k7c%28VS.85%29.aspx], it has complete reference for JScript [http://msdn.microsoft.com/en-us/library/hbxc2t98(VS.85).aspx].. Note that these are back-end or server side scripting, so not all functions or methods are accessible for browser/client (mimic) side scripting. For instance, file system object [http://msdn.microsoft.com/en-us/library/czxefwt8%28VS.85%29.aspx] handling only available for server side scripting.

Very often user may want to execute external program from within server script. Below is an excerpt from MSDN website that showing how to execute system calculator program. Paste this script into server script, trigger it with a tag, Calculator will be running when the value of this tag changes, either via user button click, setting or received from IO device. However the program will only run on server but not remote client.

```
var WshShell = new ActiveXObject("WScript.Shell");
var oExec = WshShell.Exec("calc");

while (oExec.Status == 0)
{
     WScript.Sleep(100);
}
```

And command line oriented application can be done via DOS batch file like below:

```
var obj = new ActiveXObject("WScript.Shell");
obj.Run("c:\\doswork.bat");
```

User may also perform their own file operation, below is another sample for server side scripting that executes SQL statement manually and save the output as CSV, user may customize accordingly and associate with the desired timer so it will execute periodically.

```
//compose your sql for desired data
var sql = "select time_stamp, tag_name, nvalue from log";
//assign to a string
var csv = "time_stamp,tag_name,nvalue\n" + getSql(sql, {dbgroup: 'mdb',outfmt: 'csv'

//File System Object operation
var fso, tf;
fso = new ActiveXObject("Scripting.FileSystemObject");
tf = fso.CreateTextFile("c:\\testfile.txt", true);

tf.Write ( csv);
tf.Close();

debugString( "File is saved.");
```

Note

Avoid to turn on ActiveX support unncessary as it will consume more system resources.

Chapter 5. Front End Configuration

Since IntegraXor is a standard web server, the front end is obviously built based on HTML code. This can be a little learning curve for engineer who has not exposed himself or herself to web technologies. As such IntegraXor has built in a lot of code snippets that require very minimal or totally no coding for many standard interfaces.

One good news about web content is it can be built from a huge range of technologies depending on user's level and need. So beginner could choose basic function or feature before exposing to more advance techniques. And the learning resource is so widely available, any interesting web page can be a learning lesson or example by looking into its source. And this standard information is vastly available.

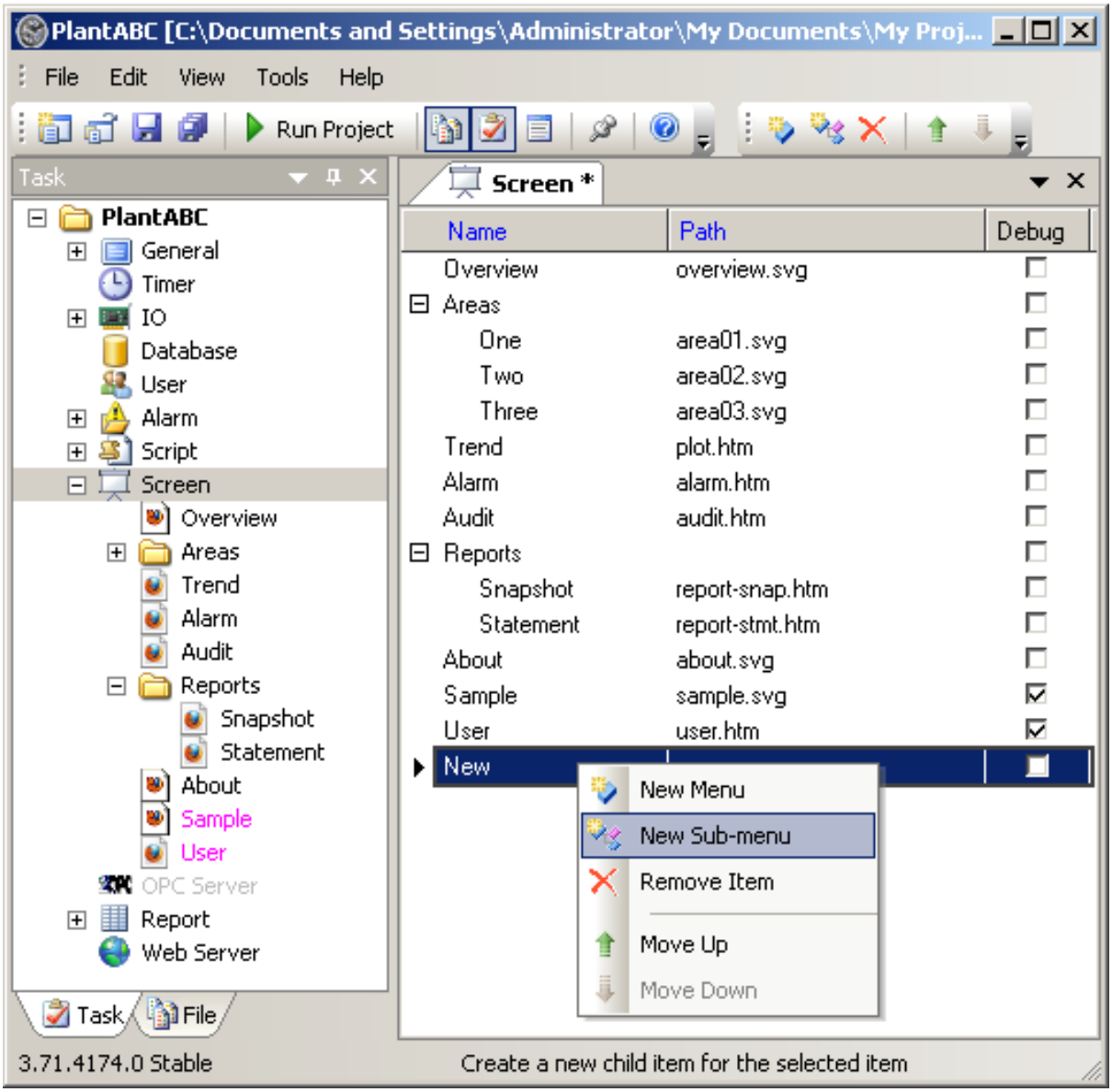
Although the knowledge require to produce web page could be higher than ordinary HMI/SCADA system's front end. However, once the basic knowledge is acquired, engineer shall find that they gain more power to accomplish task in web editing. Moreover, they will find that they can achieve objective easier and faster as compare to ordinary HMI/SCADA. Most importantly, they are extremely reliable.

Basic HTML information is widely available, this document is however assumes reader has basic knowledge of HTML and mainly discuss about the concepts of implementation. Should reader require help on basic web technologies, do contact our technical support for further reading.

There are many free and excellent HTML editing tools that are available in the market. Engineer is free to opt for any free text editor like Notepad ++ [http://notepad-plus.sourceforge.net] which can work along with Portable Apps [http://www.portableapps.com] that allows engineer to use it in a thumb or pocket drive without the need of installing it.

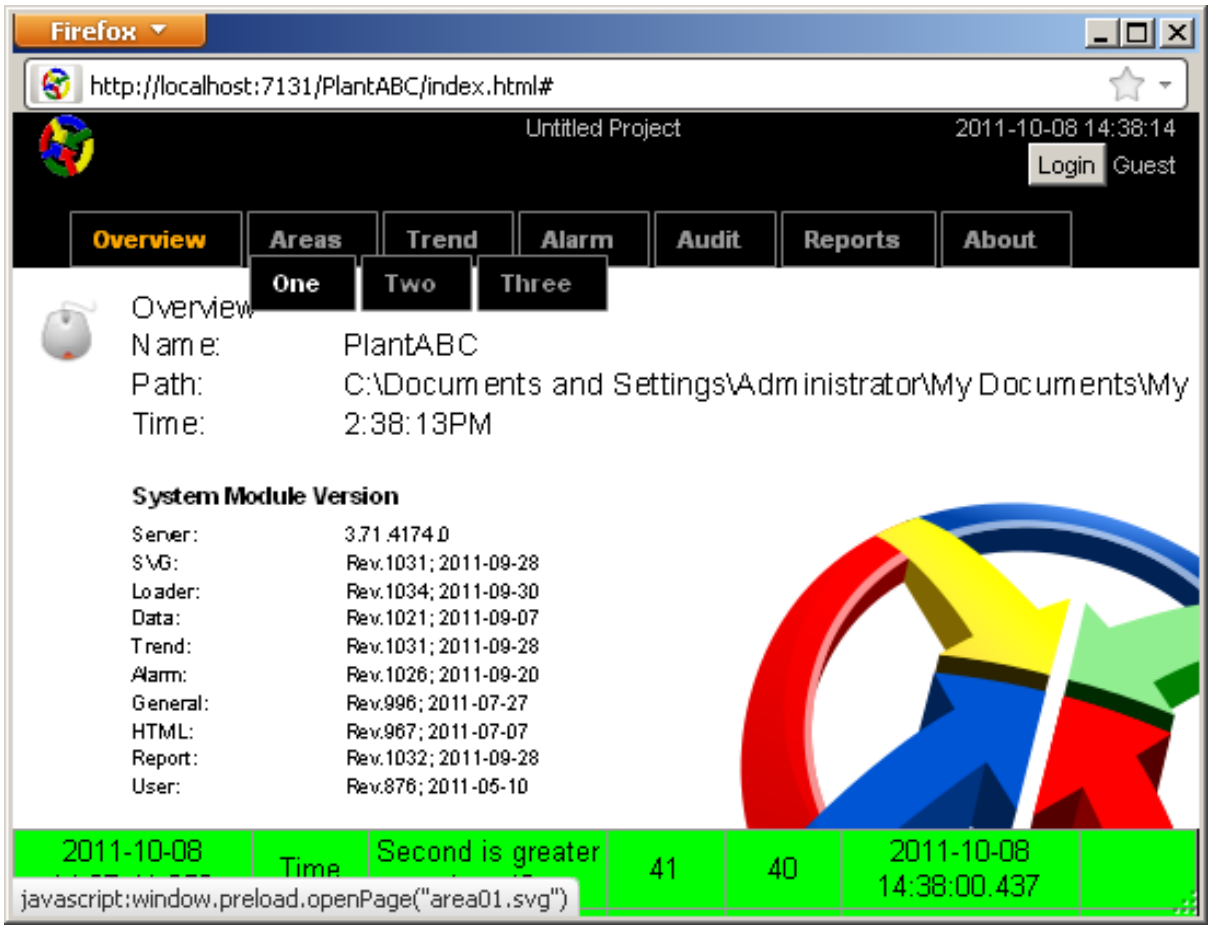
5.1. Menu

After a mimic screen in SVG/HTM formats are created, it can be link to the menu by open up Screen task. New menu/submenu can be added by right clicking at the desired position. It can then be moved up/down if needed. New screen can be added in the form of svg or htm format and associated to the menu accordingly. Moreover, any one of the screen can be set to debug status during development which will only be shown when server is set to debug mode.



Edit menu in Project Editor

Layout of SCADA screen are automated based on the page configuration above. All pages will be pre-loaded into memory, and thus page switching can be quick.



Menu is automatically generated during runtime

Note

To add new mimic page, open up Windows Explorer to create any desired page in SVG or HTML by duplicating the existing template file.

5.2. Scripting and Tag Access

igrx; is mainly developed based on standard web scripting language JavaScript or EcmaScript, although it may work with VBScript, but it's restricted to I.E. only. Engineers are encouraged to use JavaScript since it is a cross browser language. Anyhow, like Java language, C++, C# and many others, JavaScript is just another C-like language which formed by the most concise format and structure, and it is shortest, simplest and most efficient way to program. New comer may feel not comfortable at first but will soon be able to appreciate it and find other format of languages lengthy and redundant.

To access a tag value from igrx; Server, the following line needs to be included into HTML page header section, this is a very essential code snippet to be included into most HTML page that need to communicate with Server.

```
<script type="text/javascript" src="./system/scripts/igrX.js"></script>
```

Engineer only need to type `var data = getTag( 'tagname')` to read or type `setTag( 'tagname', data)` to write.

Engineer need to be aware that client (front end) tag access is bound by security checking, if any tag's read or

write access is set higher than the default security level, then end user need to login in order to read or write according to the tag's security level.

Important

The project folder is the web root directory, so the parent folder cannot be accessed from the web browser, so a file path or URL started by "./../" or "../" is not accessible.

5.3. JSON Syntax

IntegraXor code snippets use a lot of JSON (JavaScript Object Notation) [http://www.json.org] formation. Excerpt from the web site says it all: "It is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate."

Engineer do not need to know the details about it, the main message in this section is to ensure engineer could memorize a simple structure of JSON formation.

```
{'param1': data1, 'param2': data2, 'paramN': dataN}
```

As the structure above indicates, it's a simplest form ever to type, just begin the syntax with a open curly braces and always end it with a close curly braces. And the parameter and data set is always split with a colon, yet more parameters is simply separated by a comma.

Some data may consists of a list of information, then they can be represent in the following array format: `[data1, data2, data3]`, therefore if paramN above is in array format, then the whole syntax shall looks like:

```
{'param1': data1, 'param2': data2, 'paramN': [dataN1, dataN2, dataNM]}
```

Note

Just like common string data, if any of the data type above is a string, then it must be placed within a pair of single quote or double quote. For instance `{'param1': 'this is string', 'param2': 123.45}`.

Caution

Be aware that last item must not ended with comma. All quotes must be accurately matched, in another word, the syntax must be 100% comform to JavaScript language. Also, current release do not support symbol characters like &.

5.4. Number Formating

Basic number formating is supported in both SVG and HTML files. Standard numbering symbol like hash (#), zero (0), comma (,) and period (.) can be used in formatting a number. Moreover, IntegraXor number formatting is very flexible, any non-digit symbol or even space can be used as decimal or separator to support localized number formating. And the group separation can be a standard 3 digits thousand separator, or 2 digits, or 4 digits or any biggger numbers separator. Below are some examples of formating and its outcome.

Number	Format	Result
1234567.123456	`#.##`	`1234567.12`
1234567.123456	`#,###.####`	`1,234,567.1235`
0.12345	`#.##`	`.12`
0.1	`0.###`	`0.1`
1234567.890	`# ###,##`	`1 234 567,89`
1234567.890	`#.###,##`	`1.234.567,89`
1234567.890	`#,####.##`	`123,4567.89`
1234567.890	`#,##`	`1234567,89`
1234567.890	`#'###.##`	`1'234'567.89`

Tip

Entering only one symbol will always be treated as Decimal, simply add a trailing space at the end to force it become Separator. For instance, inputting 1234567.890 into `"#,### "` will generate `1,234,568`.

5.5. Tabulate Data

HTML page is natively excellence in tabulating data. Data could be well aligned and displayed in HTML page easily without much effort. Unlike ordinary HMI/SCADA, data might be out of alignment and displayed with difference sizes of text. Although there's visual editing tool to create HTML table, engineer is encourage to go for hand coding creation as they will gain full control in styling the table and avoid unnecessary codes.

No coding is needed for tabulating data in HTML page, like previous Tag Access, engineer only needs to include the following script library:

```
<script type="text/javascript" src="./system/scripts/igrX.js"></script>
```

To show a tag value to a table cell, name the ID of the particular cell as `{attr:'get', tag:'tagname'}`.

Tip

When a table cell is animated with a tagname, it's recommended to fill up the cell with a Non-Breaking Space, i.e. type ` ` into the cell. This shall prevent table rows from adjusting or expanding itself when data is received right after the web page is initially loaded without data.

5.6. Printing

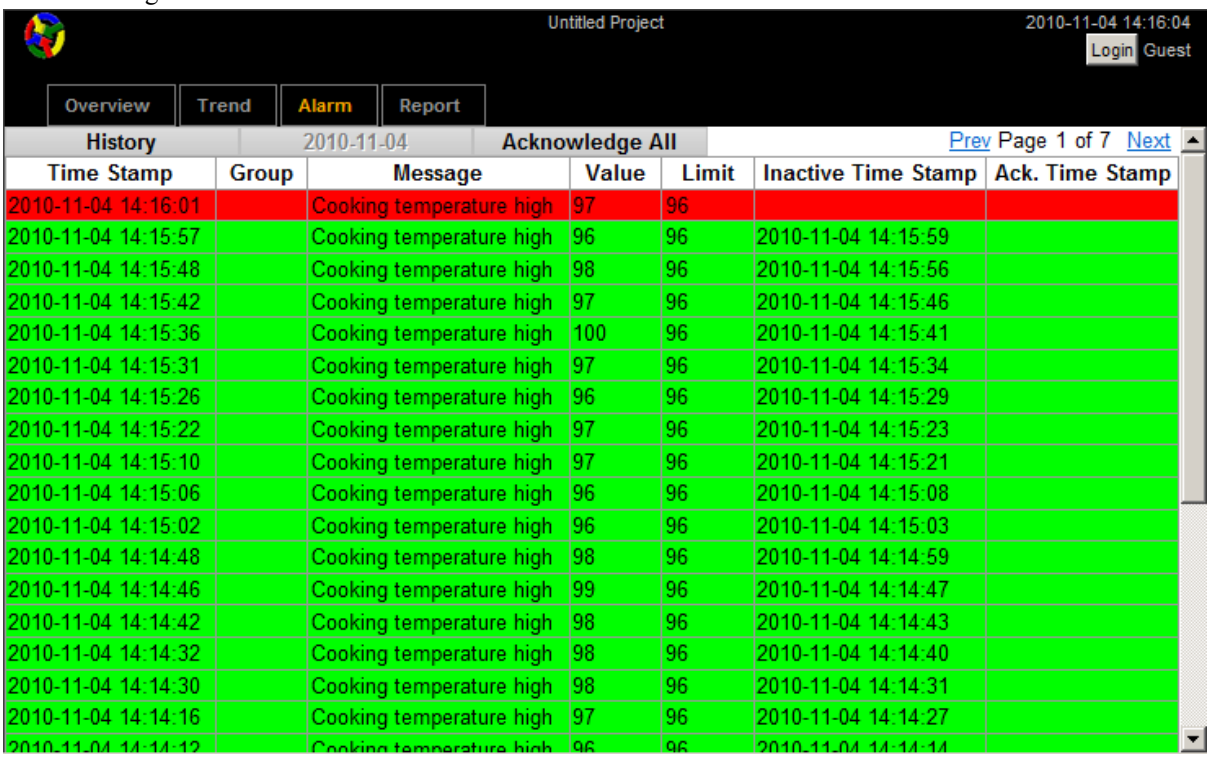
Printing is again in advantage in HTML work just like data tabulation. Report generation or printing for many ordinary HMI/SCADA can be a nightmare, not only it could be difficult to generate but yet could be unreliable. Unlike other HMI/SCADA system, IntegraXor leveraged on web browser that can easily have the displayed data printed on client side. One big advantage of HTML printing is it's truely wysiwyg, this simple old day requirement is still severely lacking in almost all other HMI/SCADA systems, if not they may require engineer purchase additional plug-in like Crystal Report or Microsoft Excel that introduce additional yet repetative cost.

Note that all printing at client side, i.e. via browser shall invoke a standard printing dialog box so user could choose to print to the desired printer and several other standard printing options. As such automatic printing without user intervention is not possible via HTML code. Some of the I.E. version installed on Windows comes with security loop hole that allow user to bypass this standard printing dialog box with some tricks. However, this is no longer possible with newer version of I.E. and Windows security patches. Engineer is discouraged to design their project based on this loop hole.

IntegraXor provides built-in function `print('url', DELAY, 'nu')` so that printing without dialog box is possible. Note that this printing is triggered from Server, so it can only print to printer that within its network reach. More details is described at Logic or Scripting section.

5.7. Alarm Viewer

Alarm Viewer work in conjuction with alarm configuration at server, yet alarm require database to be running so that alarm can be logged and fetched. Alarm Viewer HTML page is readily portable and can be plug into any project without any modification. Engineer could simply link Alarm Viewer 'alarm.htm' to anywhere in the project, or just use the default layout provided in the project template. Below is its pretty self-descriptive buttons' functions and available features. The color legend also very intuitive, red/pink showing Active alarm, yellow/wheat showing Active and acknolwedged alarm, and lastly green/lime showing Non-active but unacknowledged alarm.



Time Stamp	Group	Message	Value	Limit	Inactive Time Stamp	Ack. Time Stamp
2010-11-04 14:16:01		Cooking temperature high	97	96		
2010-11-04 14:15:57		Cooking temperature high	96	96	2010-11-04 14:15:59	
2010-11-04 14:15:48		Cooking temperature high	98	96	2010-11-04 14:15:56	
2010-11-04 14:15:42		Cooking temperature high	97	96	2010-11-04 14:15:46	
2010-11-04 14:15:36		Cooking temperature high	100	96	2010-11-04 14:15:41	
2010-11-04 14:15:31		Cooking temperature high	97	96	2010-11-04 14:15:34	
2010-11-04 14:15:26		Cooking temperature high	96	96	2010-11-04 14:15:29	
2010-11-04 14:15:22		Cooking temperature high	97	96	2010-11-04 14:15:23	
2010-11-04 14:15:10		Cooking temperature high	97	96	2010-11-04 14:15:21	
2010-11-04 14:15:06		Cooking temperature high	96	96	2010-11-04 14:15:08	
2010-11-04 14:15:02		Cooking temperature high	96	96	2010-11-04 14:15:03	
2010-11-04 14:14:48		Cooking temperature high	98	96	2010-11-04 14:14:59	
2010-11-04 14:14:46		Cooking temperature high	99	96	2010-11-04 14:14:47	
2010-11-04 14:14:42		Cooking temperature high	98	96	2010-11-04 14:14:43	
2010-11-04 14:14:32		Cooking temperature high	98	96	2010-11-04 14:14:40	
2010-11-04 14:14:30		Cooking temperature high	98	96	2010-11-04 14:14:31	
2010-11-04 14:14:16		Cooking temperature high	97	96	2010-11-04 14:14:27	
2010-11-04 14:14:12		Cooking temperature high	96	96	2010-11-04 14:14:14	

Alarm Viewer

Alarm Viewer is customizable by changing the configuration in *alarm.htm*.

```
var opt = {
    "mode": "alarm",
    "row-per-page": 30,
    "audio": "media/ding.wav",
    "column": {
        "time_stamp": true,
        "tag_name": false,
```

```
            "group_name": true,
            "description": false,
            "message": true,
            "nvalue": true,
            "old_nvalue": true,
            "inactive_timestamp": true,
            "ack_timestamp": true,
            "ack_user": false
        }
    };
```

In the above configuration, alarm viewer can be configured into 3 modes, i.e, "alarm", "footer", and "audit". Alarm mode is the normal alarm mode with acknowledge and date selection function. Footer mode is similar to alarm mode, except it has no control panel (date selection, real time/history mode). Audit mode will be described in next page.

"row-per-page" option stands for the number of item/row the alarm viewer shall display. The column headers can be customized by specifying *true/false*. Total number of alarm to display per page is also configurable.

Column Header	ID	Description
Time Stamp	time_stamp	This column shows the timestamp of alarm
Tag	tag_name	Tag of the alarm
Group	group_name	Grouping based on configuration
Description	description	Description of the alarm
Message	message	Alarm message string
Value	nvalue	Value stamped during occurrence of alarm
Limit	old_value	Configured alarm limit for reference
Inactive Timestamp	inactive_timestamp	Timestamp of inactive
Acknowledge Timestamp	ack_timestamp	Timestamp of alarm acknowledgement
User Acknowledge	ack_user	User who acknowledge the alarm

Tip

To change the column/header of footer, open `index.html` instead of alarm.htm, and locate the configuration.

To override the system default color, paste the following into `<header>` section and customize the color accordingly.

```
<style type="text/css">

.alarm-table .active {
    background-color: blue;
}
.alarm-table .inactive {
    background-color: pink;
}
.alarm-table .active-ack {
    background-color: brown;
}
.alarm-table .inactive-ack {
    background-color: purple;
```

```
}
</style>
```

Tip

User can show only alarm with specific conditions (a.k.a filters) such as group name or tag name. This setting allows user to have multiple alarm viewer for different groups.

```
var opt = {
    "mode": "alarm",
    "row-per-page": 30,
    "column": {
        "time_stamp": true,
        "tag_name": false,
        "group_name": true,
        "description": false,
        "message": true,
        "nvalue": true,
        "old_nvalue": true,
        "inactive_timestamp": true,
        "ack_timestamp": true,
        "ack_user": false
    },
    "filter": {
        "group_name": "Generator"
    }
};
```

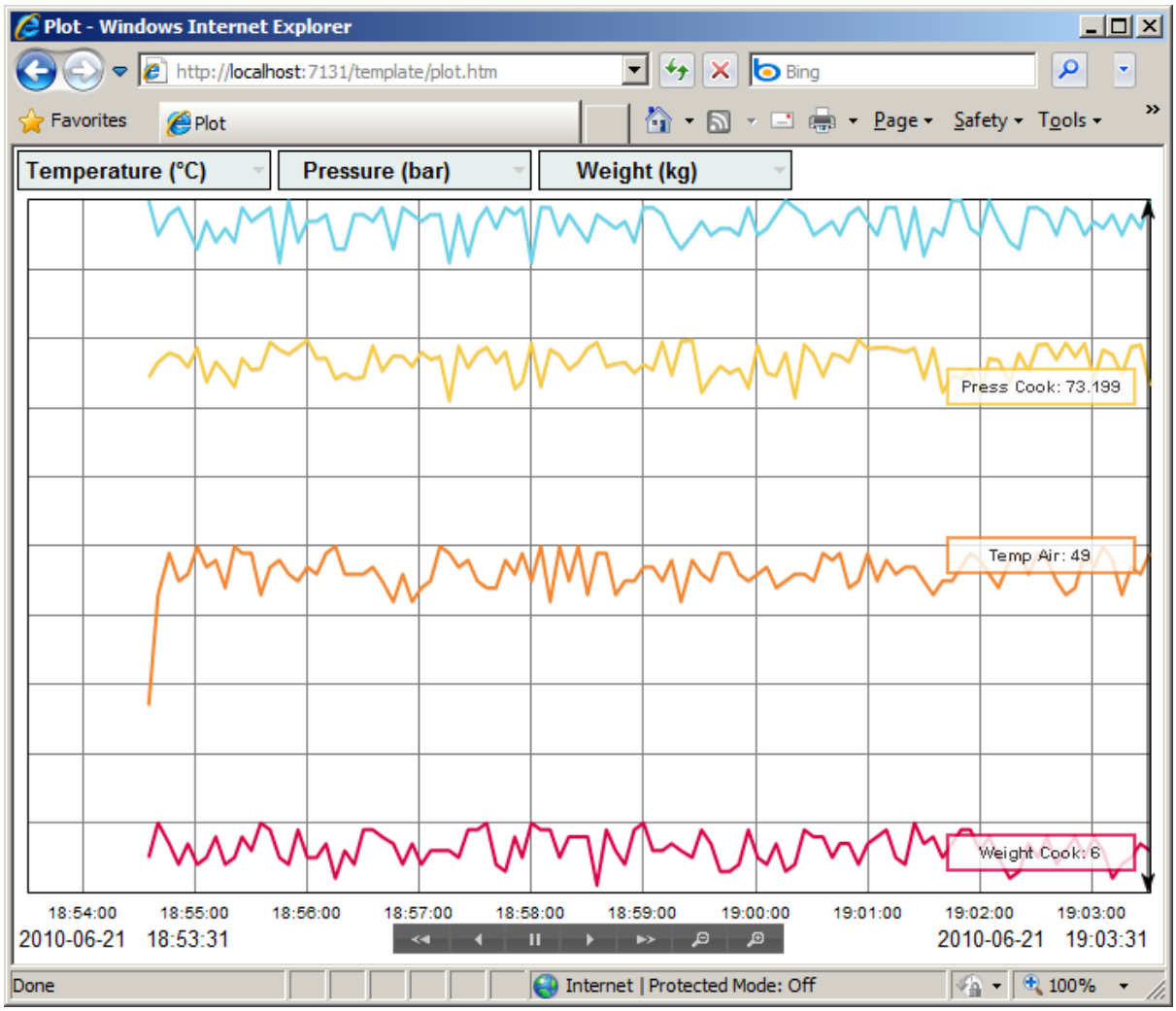
5.7.1. Audit

Audit, also known as event logging, is a sub-module of alarm viewer. To enable audit mode, change the JSON configuration *mode* from *alarm* to *audit* as below:

```
var opt = {
    "mode": "audit",
    "row-per-page": 30,
    "column": {
        "time_stamp": true,
        "tag_name": false,
        "group_name": true,
        "description": false,
        "message": true,
        "nvalue": true,
        "old_nvalue": true,
        "inactive_timestamp": true,
        "ack_timestamp": true,
        "ack_user": false
    }
};
```

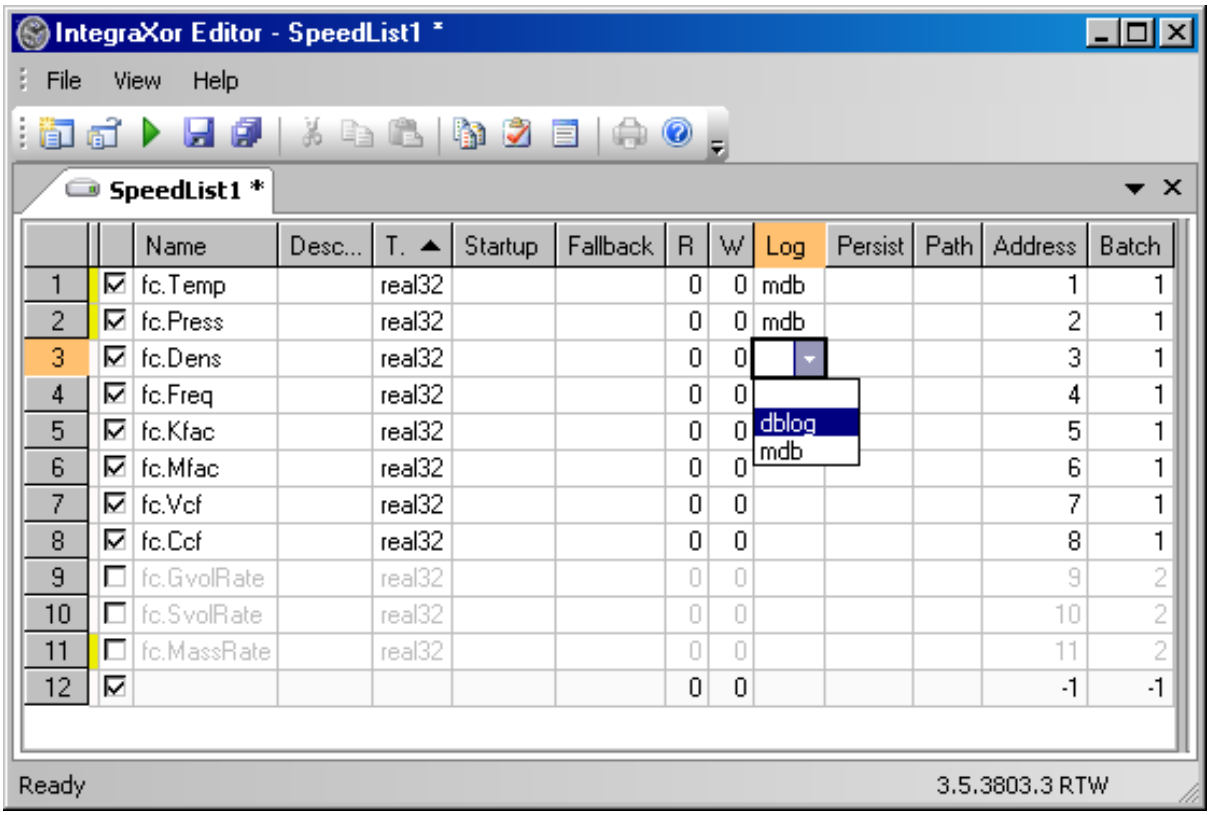
Audit mode has real time and historical mode as well. In real time mode, all events will be shown on screen with detail of their last activity. In historical mode, it will show all the past activities in selected date.

5.8. Trending



General Plot Screen

Trending requires database to be running at server. After database is configured, to plot a trend for a tag is as simple as selecting the database for the desired tag to plot.


IntegraXor Editor - SpeedList1 *
File View Help
SpeedList1 *
Name Desc... T. Startup Fallback R W Log Persist Path Address Batch
1 fc.Temp real32 0 0 mdb 1 1
2 fc.Press real32 0 0 mdb 2 1
3 fc.Dens real32 0 0 3 1
4 fc.Freq real32 0 0 4 1
5 fc.Kfac real32 0 0 5 1
6 fc.Mfac real32 0 0 6 1
7 fc.Vcf real32 0 0 7 1
8 fc.Ccf real32 0 0 8 1
9 fc.GvolRate real32 0 0 9 2
10 fc.SvolRate real32 0 0 10 2
11 fc.MassRate real32 0 0 11 2
12 0 0 -1 -1
dblog
mdb
Ready
3.5.3803.3 RTW

Select a tag to log into database

Trending can be initiated by javascript function called *trend()* within an HTML file with a single DIV tag. Below is the complete snippet including configuration that will load a trending screen.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <link rel="stylesheet" media="screen" href="system/styles/igrX.css"/>
</head>
<body>
    <div class='trend' id='placeholder'></div>
</body>
<script type="text/javascript" src="system/scripts/igrX.js"></script>
<script type="text/javascript">

var pen = [
    {
        name: 'Temperature',
        enabled: true,
        unit: 'Celsius',
        min: 0,
        max: 100,
        pens: [
            {
                name: 'Temperature Cook',
                tag: 'temperature_cook'
            }
        ]
    }
];
var opt = {
    lograte: 5000,
    backgroundColor: 'white'
};

var plot = new trend( '#placeholder', {
```

```
    'pen_setting': pen,
    'option': opt

});

</script>
</html>
```

There are two arguments required: DIV to append and setting. The function call will append the chart to a DIV object with ID "placeholder".

```
var plot = new trend( "#placeholder", {
    "pen_setting": pen,
    "option" : option
});
```

Setting is a JSON object consists of 2 child objects, called `pen_setting (Array)` and `option (JSON)`. In pen_setting, user is required to enter basic configuration of the pens that need to trend. A sample of pen_setting configuration for Temperature and Pressure is as per below:

```
var pen = [
    {
        name: "Temperature",     // (String)
        enabled: true,           // (Boolean)
        unit: "°C",              // (String)
        min: 0,                  // (Number)
        max: 100,                // (Number)
        pens: [
            {
                name: "Temp A",  // (String)
                tag: "temp_a",   // (String)
            },
            {
                name: "Temp B",  // (String)
                tag: "temp_b",   // (String)
            },
            {
                name: "Temp C",  // (String)
                tag: "temp_c",   // (String)
            }
        ]
    },
    {
        name: "Pressure",        // (String)
        enabled: true,           // (Boolean)
        unit: "kPa Abs",         // (String)
        min: 0,                  // (Number)
        max: 10,                 // (Number)
        pens: [
            {
                tag: "press_a",  // (String)
            }
        ]
    }
];
```

Each element in pen_setting is a parameter to a number of tags. For each parameter, user is required to enter details for the particular parameter.

Field	Data Type	Description
name	String	Name of the parameter, e.g. Temperature.
enabled	Boolean	To show(true) / hide(false) the parameter upon startup. Default to true if not specified.
unit	String	Unit of the parameter, e.g. Celsius.
min	Number	Minimum value of the parameter.
max	Number	Maximum value of the parameter.
setPoint	String	Tag name of the set point. Value of the tag will be taken as set point. Set point is a horizontal reference line for a parameter.
setPointColor	String	Color for the set point horizontal reference line.
pens	Array	Contains the pens under this parameter.

Pens is an Array consists of JSON objects, while each of the JSON objects is configuration of a pen.

Field	Data Type	Description
name	String	Label of the pen. Default to tag if not specified.
tag	String	Tag of the pen.
enabled	Boolean	To show/hide the pen on startup. If it is not specified, it will inherit from parameter setting.

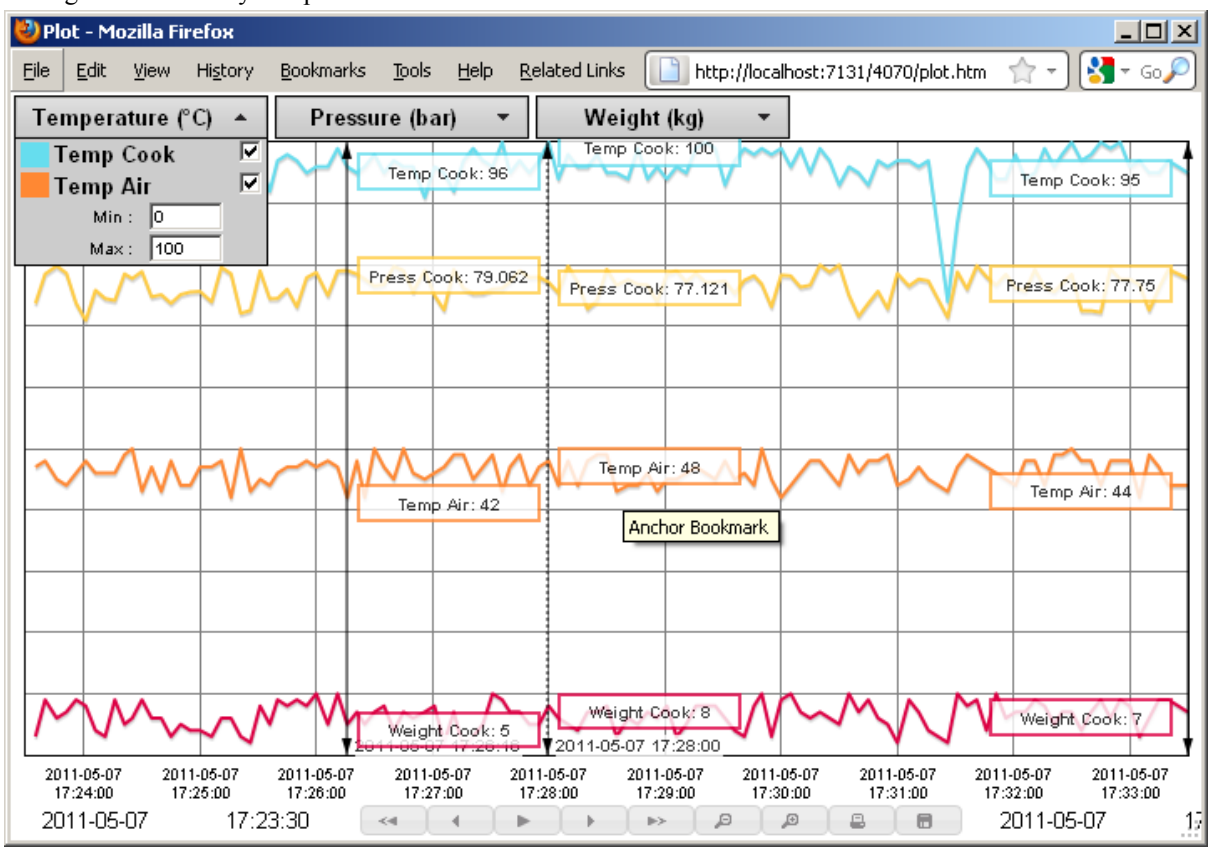
Second object is called *option*. Option is where the user set various general setting to the chart. Below is an example:

```
var option = {
    lograte: 5000,
    backgroundColor: "white"
};
```

Field	Data Type	Description
gridx	Number	Number of X-axis grid to display. Default to 10 if not specified.
gridy	Number	Number of Y-axis grid to display. Default to 10 if not specified.
legend	Boolean	To show legend on top right corner of the graph. Default to false if not specified.
length	Number	Startup length of the plot in term of time span. Default to 600 (second) if not specified.
realTime	Boolean	Startup status of trend. Default to true if not specified.
lograte	Number	Log rate of the database in millisecond. Should be same as logging rate of database.
source	String	Define the database table to plot. Default to "log" table if not specified.
dbgroup	String	Define the name of database where the source is. This is reserved for report module. Default to "dblog" if not specified.
skip	Boolean	To allow the trending system to skip data from database during each query, to display optimum number of points on screen. This is to prevent

Field	Data Type	Description
		database from overloading. Default to false if not specified.
breaklimit	Number	Define how many millisecond between 2 coordinates should consider missing data. If the time between 2 coordinates are greater than this value, line will not be drawn. Default to 2x lograte if not specified.
backgroundColor	String	Back-ground color of chart. Default to white if not specified.
refLineColor	String	Reference line color of chart. Default to black if not specified.

All pens will be plotted in a graph, with respect to their own maximum/minimum configuration. Maximum/minimum value of the parameter, as well as color of the individual pens can be seen on a drop down window when the parameter on top panel is clicked. Maximum and minimum of a parameter can be changed by editing the text directly and press Enter.



Plot Screen

User can click on the graph to add a reference line. Any pen that intercept with the line will have a tooltip of value displayed. There are 2 types of reference line: Bookmark and Anchor. Bookmark will stick with the time stamp and eventually drift away in real time trending, while anchor will stay on the screen. When user clicked on the graph to add a reference line, a bookmark will be created (dotted line). User can convert the bookmark to an anchor by clicking on it once (static line). By clicking on the anchor again, the reference line will be removed.

Navigation panel is located at the bottom. Buttons from left: Pan Left by Page, Pan Left by Grid, Pause/Play, Pan Right by Grid, Pan Right by Page, Zoom out, Zoom in. Other than using buttons, user can select specific time stamp by clicking on the date/time at bottom left or bottom right corner. A datepicker/timepicker will prompt for selection. If any end date/time earlier than a begin date/time is selected, the graph will do nothing but turn to historical mode. Ultimately, user can click and drag on any specific portion for zooming in.

Note

Any button clicked, or date/time changed will result in historical mode. Operator will need to click on "Play" button in order to resume in real time mode.

Advanced Javascript user can utilize the trend object created by calling the public functions of trend. There are currently 2 public functions exposed, i.e. *setTimeStamp()* and *realTime()*. setTimeStamp() takes in 1 parameter in Object format, where the element "bgn" is UNIX format of begin time while the element "end" is UNIX format of end time. realTime() takes in 1 boolean parameter, to force the graph into real time or historical mode. Calling realTime() without parameter will toggle between the 2 modes.

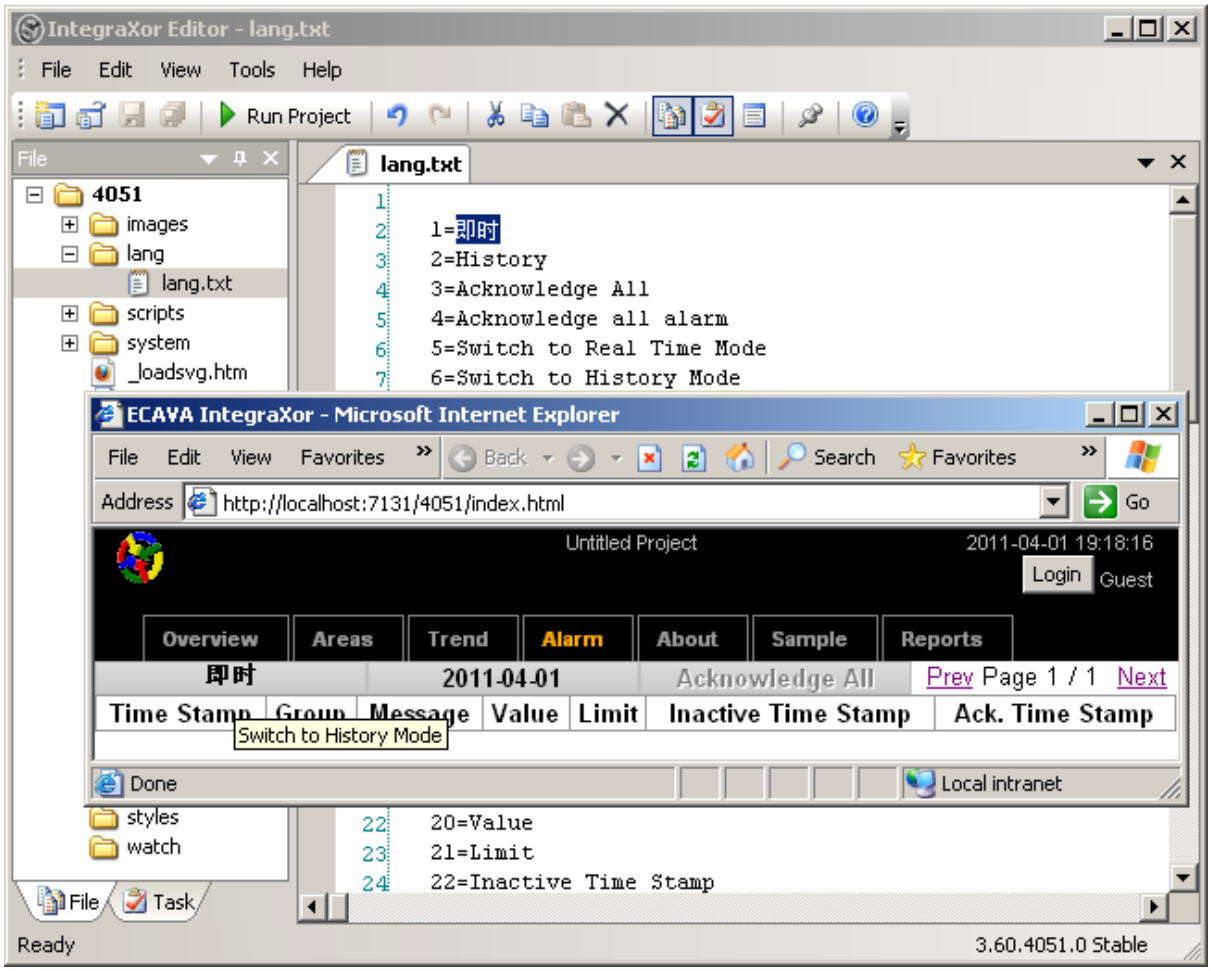
```
var new_time = {
    "bgn": 1270000000000,
    "end": 1271000000000
};
plot.setTimeStamp(new_time);


plot.realTime(false);
```

Tip

An alternative method in placing the DIV other than using ID is using `classname` as reference. The parameter is a CSS selector of DOM, i.e, "#" represents ID and "." represents class name. So to use `classname` as reference, supply `".placeholder"` as first parameter for `trend()` function call.

5.9. Multilingual support

User can translate all frontend display texts into local language by editing the `lang.txt` file which resides project folder sub-directory `/lang` folder. The text file shall be saved in UTF-8 format.



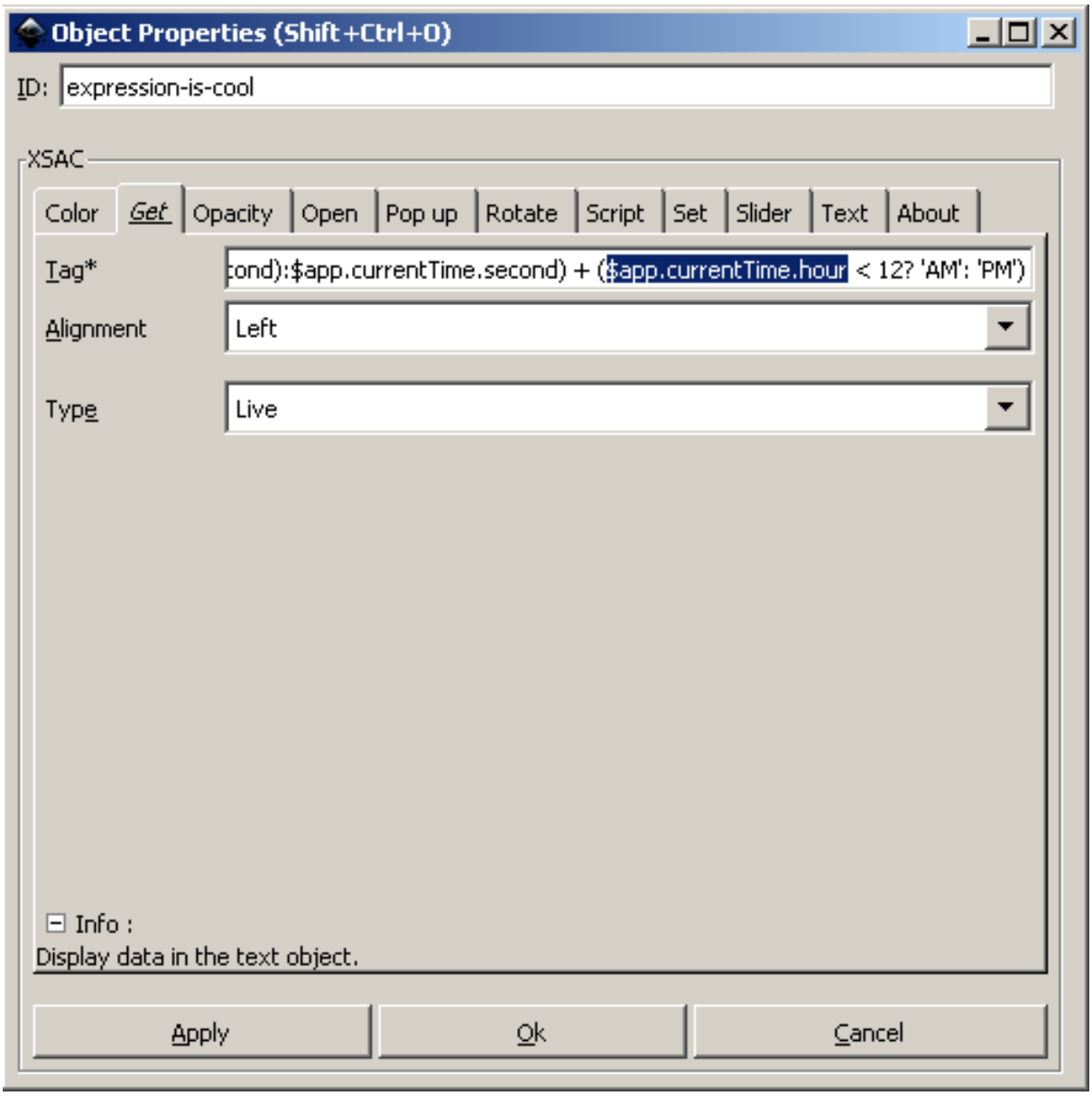
Multi lingual support for web interface.

5.10. Logical Expression

Logical Expression can be done on front-end tag field, and it is achieved by putting a leading dollar sign ($) in front of a tag name. IntegraXor supports simple mathematic calculation to full fledge complicated logic operation. Tagname must be compliant to Variable Naming convention or the result will be unpredictable, i.e. contains no operation symbol like dash or asterisk etc, and shall not start by a number. A bad example is such that a dash in a tag name will be treated as minus.

Expression for front-end is handy to be used when a tag value need to be converted to different engineering unit or shown in different format. For instance a pressure value is received as bar and being used as it is through out the project, but need to be displayed in kPa in certain places. Then instead of entering the tag name directly, user may enter an expression like this: `$tagname * 100` .

For a logical operation instance, a project requirement is such that a value need to be shown as zero at all time unless when a boolean tag has been turned on: `($tag_bool)? $tag_int: 0` . Another practical use of Expression is to display a default value when a tag is not initialized: `$tag||1.01325` .



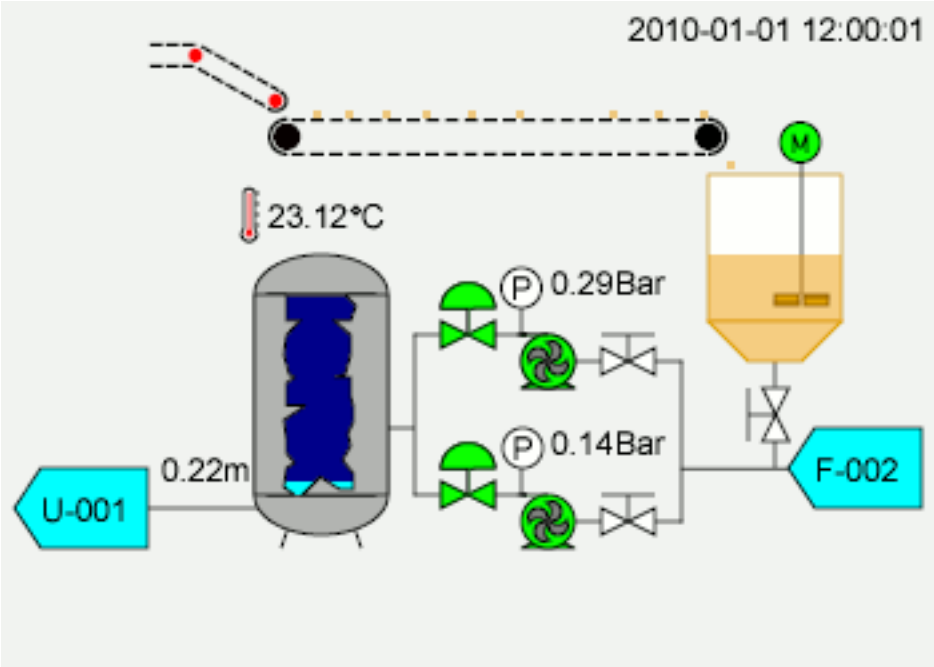
Expression being used in default template's overview.svg showing AM/PM based on current hour.

Note

Despite Expression is very convenient to be used, but it does come with price that it will impact the front-end performance in certain level, due to the nature that the logic will only be executed before viewing. So excessive use of Expression at front-end shall be avoided on low performance CPU.

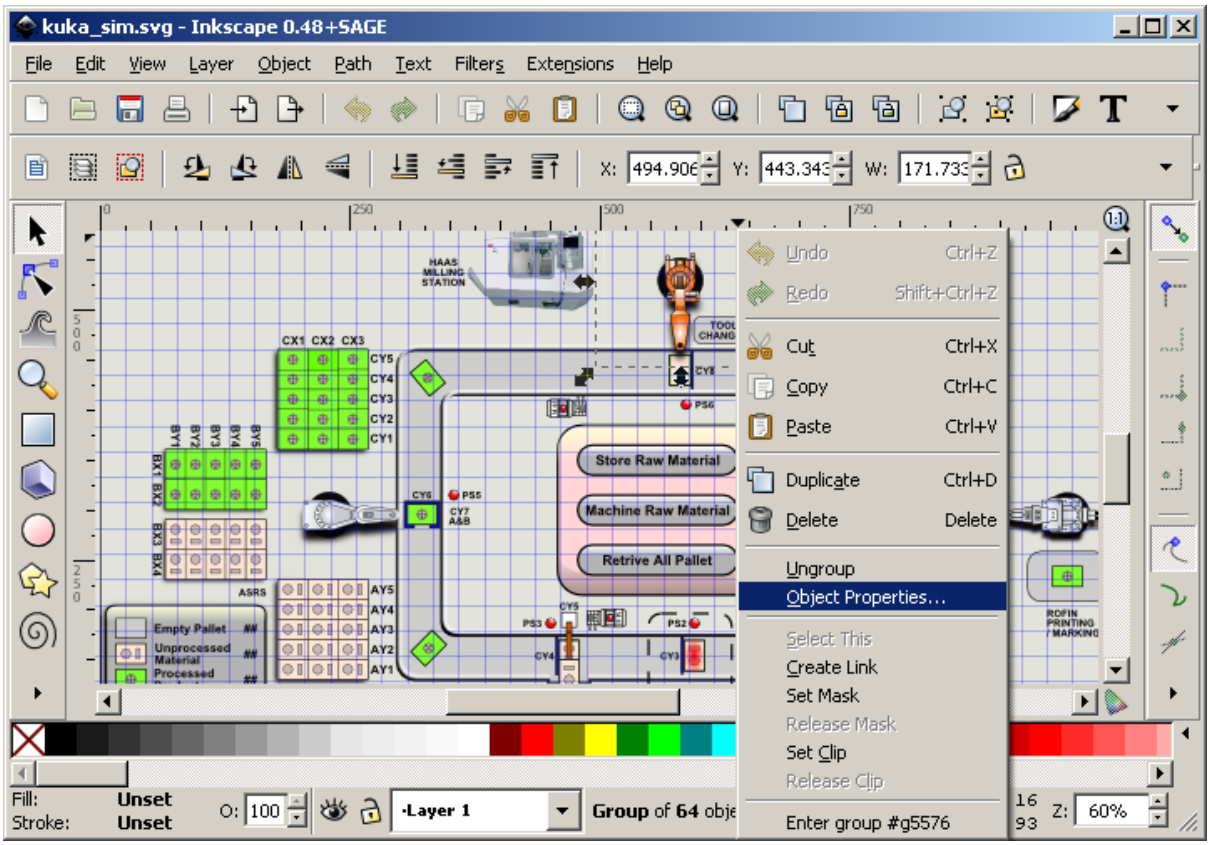
Chapter 6. Inkscape SAGE

Basic SCADA animation consists of Color Indication, Rotation, Bar (Level), Movement, Text (Data), Opacity (Hide/Show) and etc. Most SCADA applications can be achieved by using these basic animations. Below is a basic SCADA animation that can be easily configured with open source Inkscape+SAGE [http://sourceforge.net/projects/sage-scada/]. More advanced animations are available at this Interactive Real-time Internet SCADA Demo [http://www.integraxor.com/demo.htm?sl=ug].



Basic SCADA Animation Mimic

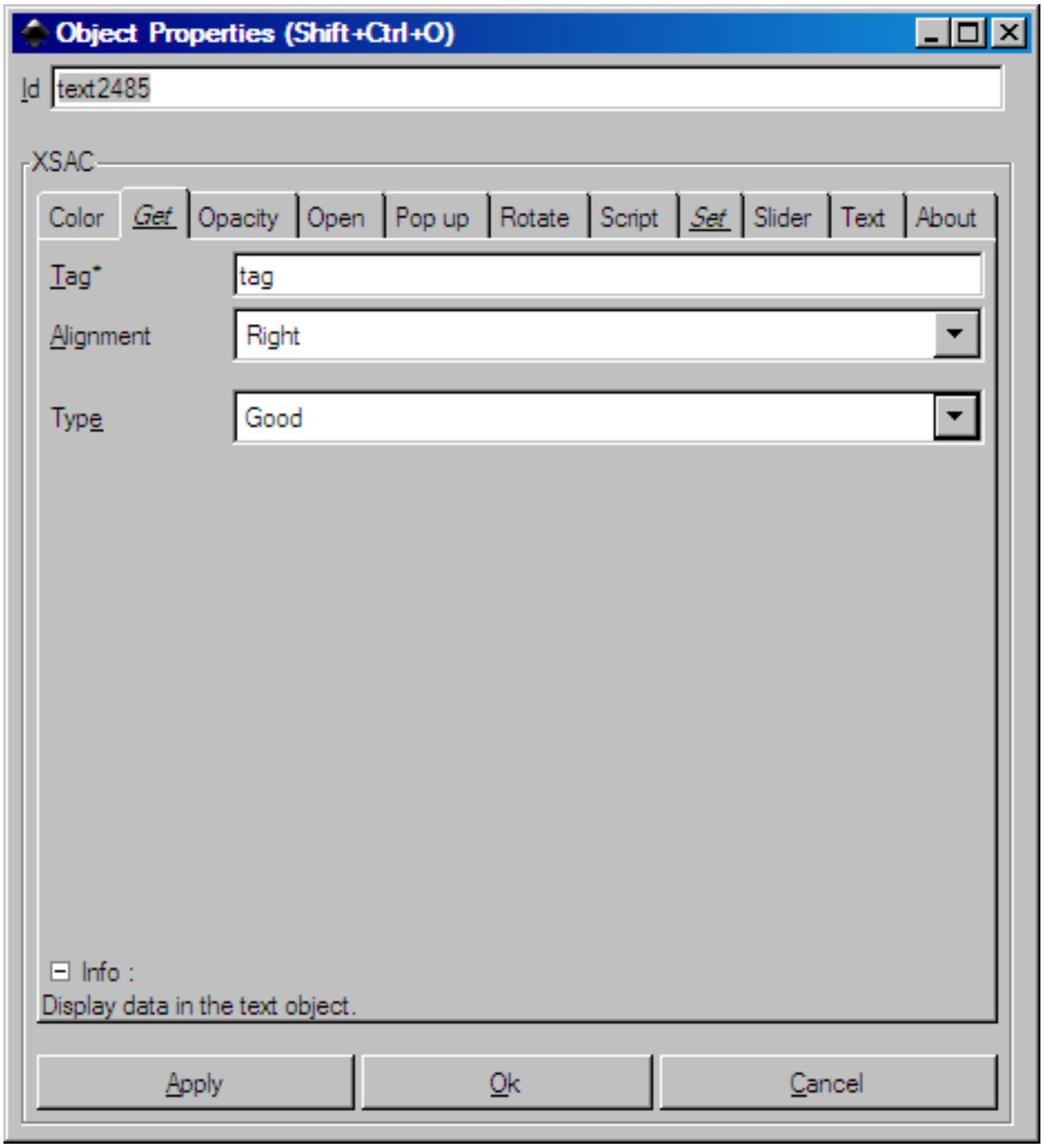
SAGE (SCADA Animation Graphic Editor) is a modified interface for Inkscape. After completed with graphics, the next task is to attach animations to them accordingly. This can be done by editing the object properties. Open object properties by highlighting the object, and right click then click on Object Properties. Alternatively, user can highlight the object and press CTRL + SHIFT + O to open up Object Properties window.



Right click to open object properties

6.1. Get Animation

Get animation is an animation that allows user to display (read) a tag value in SVG directly, either as string or number. To attach a Get animation, the SVG Object must be a text. Creating rectangle or circle will not have Get animation in Object Properties. If the value is a number, User can format the value into desired format by editing the SVG text object directly into number format (example please see Number Formating Section).



Get Animation

Figure shows the get animation of an object. Only the 'Tag' field need to be entered. Both 'Alignment' & 'Type' shall not be used as they are deprecated or reserved for future. See Text Alignment for more configurations.

6.1.1. Inkscape Text Alignment

Text alignment [http://tavmjong.free.fr/INKSCAPE/MANUAL/html/Text-Formatting.html#Text-Formatting-Justification] can be set on left, right & center within Inkscape itself via Toolbar, which will appear when entering text editing mode by double clicking on Text, or pressing F8 when the text is selected. Alternatively, text also can be aligned via Text & Font dialog by pressing Ctrl+Shift+T when text is selected.

Be aware that text will remain at its original position when aligning text via toolbar. And text will change its position when aligning via popup dialog. Either way, observe that the anchor of the text which is indicated by a

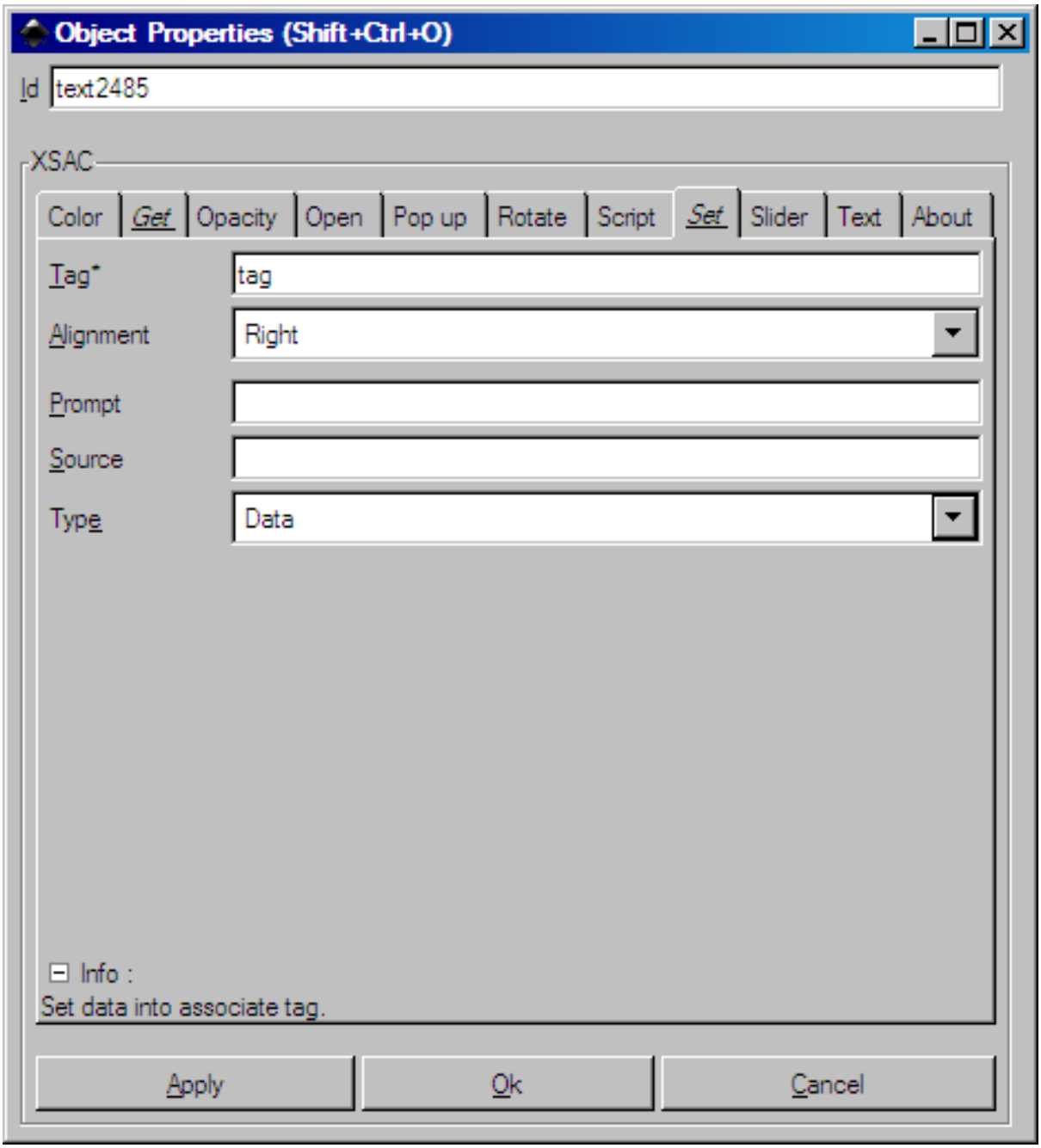
small little box at the bottom of the text will appear on the left, right & center to denote its configured alignment.

Note

Do not use deprecated text alignment in SAGE-Get animation, using alignment animation together with Inkscape built-in alignment will not yield expected alignment.

6.2. Set Animation

Set animation is an animation that allows user to write value into a tag (write) directly by clicking the SVG Object, either as string or number. Set animation is not necessary to be text, user can attach Set animation with any SVG Object.

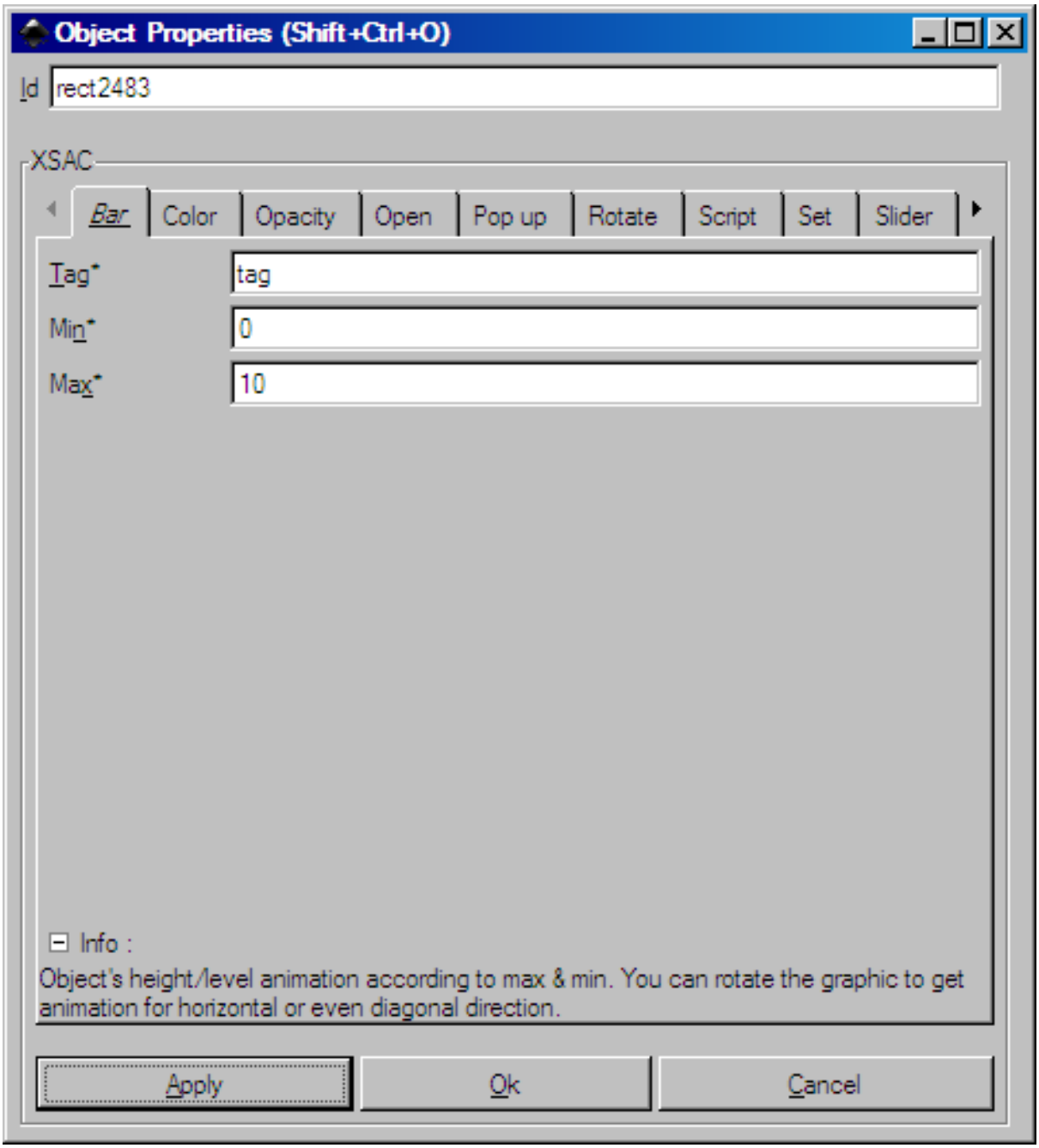


Set Animation

Figure above shows the set animation of an object. There is one mandatory input to be entered: tag. Tag will be the IO tag in IntegraXor server which the user wants to write, where alignment will determine the position of output. Prompt input is a message to be delivered to confirm changes. Type is a drop down menu of 2 choices: Data or Variable. If Type is set to Data, then Source will be the data to write into the associated tag. If Type is set to Variable, then Source will be read as a tag, where the value of the source tag will be write into the associated tag. If Type and Source is not changed (by default), the Set animation will prompt end user an edit box to change value of attached tag on demand.

6.3. Bar Animation

Bar animation is an animation where the object's height will be animated according to the minimum and maximum declared, in respect to the value of tag attached. The tag value over the difference between maximum and minimum will be the percentage of result height.



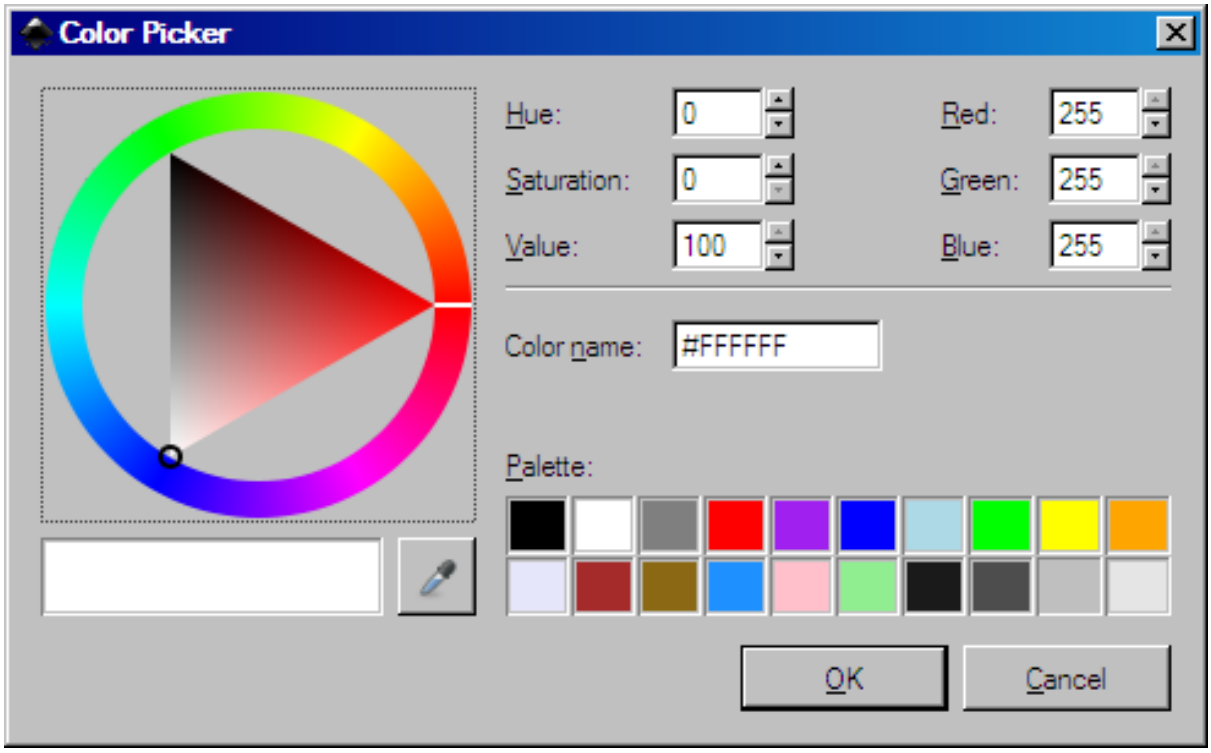
Bar Animation

Figure above shows the bar animation of an object. There are three mandatory input to be entered: tag, min and max. Tag will be the IO tag in IntegraXor server which the user wants to attach, where min is the minimum value and max is the maximum value that the object will be resize into. Click OK and exit, and the animation is added successfully. Note that in this graphic editor, user is able to rotate the graphic into any angle. Therefore, the height of object is which the object is initially created. To be more exact, open XML editor and check for more details.

6.4. Color Animation

Color animation is one of the mostly used animation, it simply display the desired color when the associated tag matches or fall within the condition. Click 'Add' to start adding animation condition by entering tag, condition and the desired color. Color can be entered by clicking the "Color Picker" button. A pop up of color picker

window will prompt for color selection. Note that the priority of the animation conditions can be re-arranged by clicking "Move up" or "Move down" button.

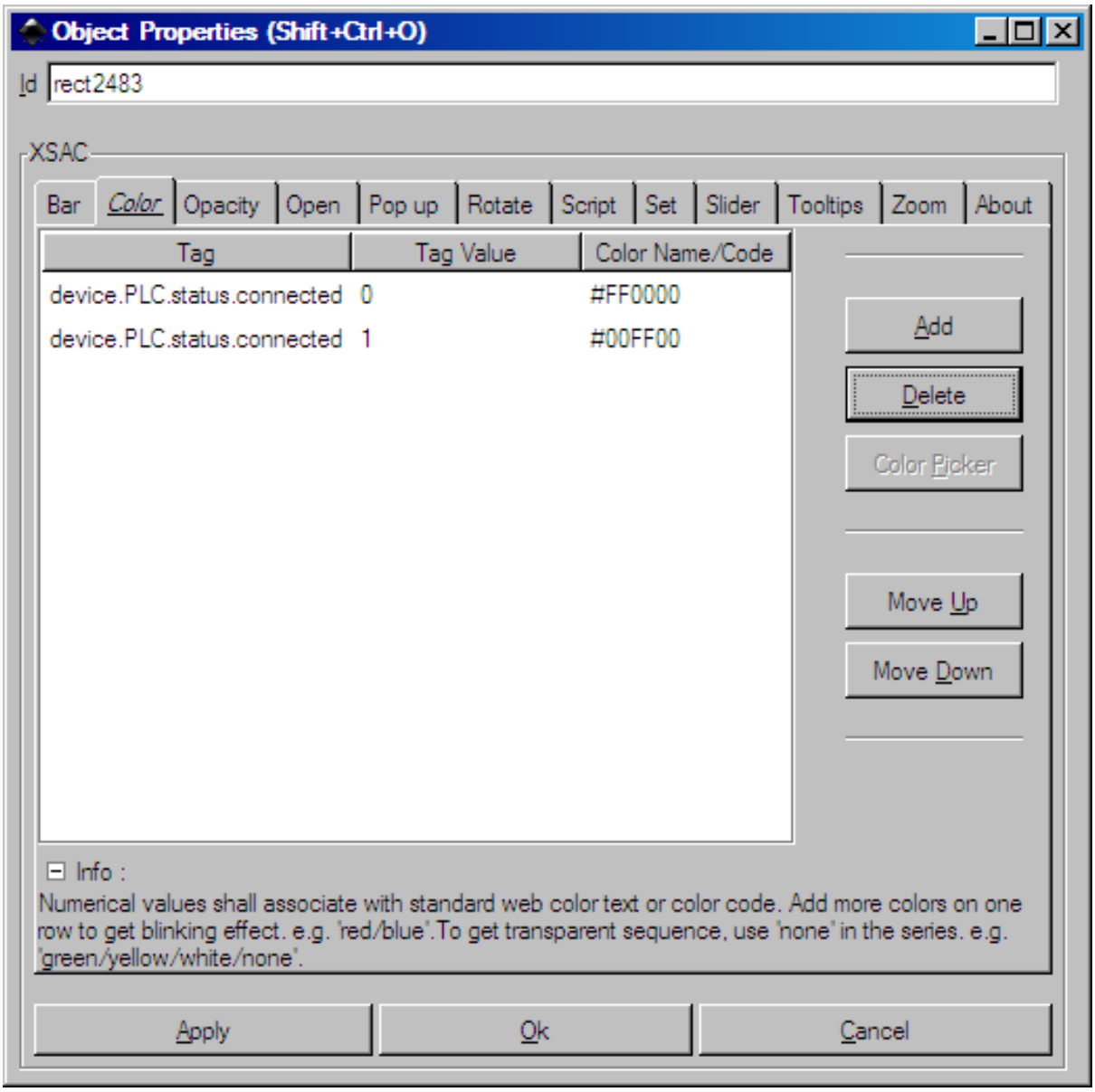


Color Picker

After selecting color, a color code will appear in color code column. Below is a sample of status tag configuration for a device connection status. Since it's a boolean tag, only 2 colors are entered. When the device is disconnected, red will be shown and when the device is detected, green will be shown instead.

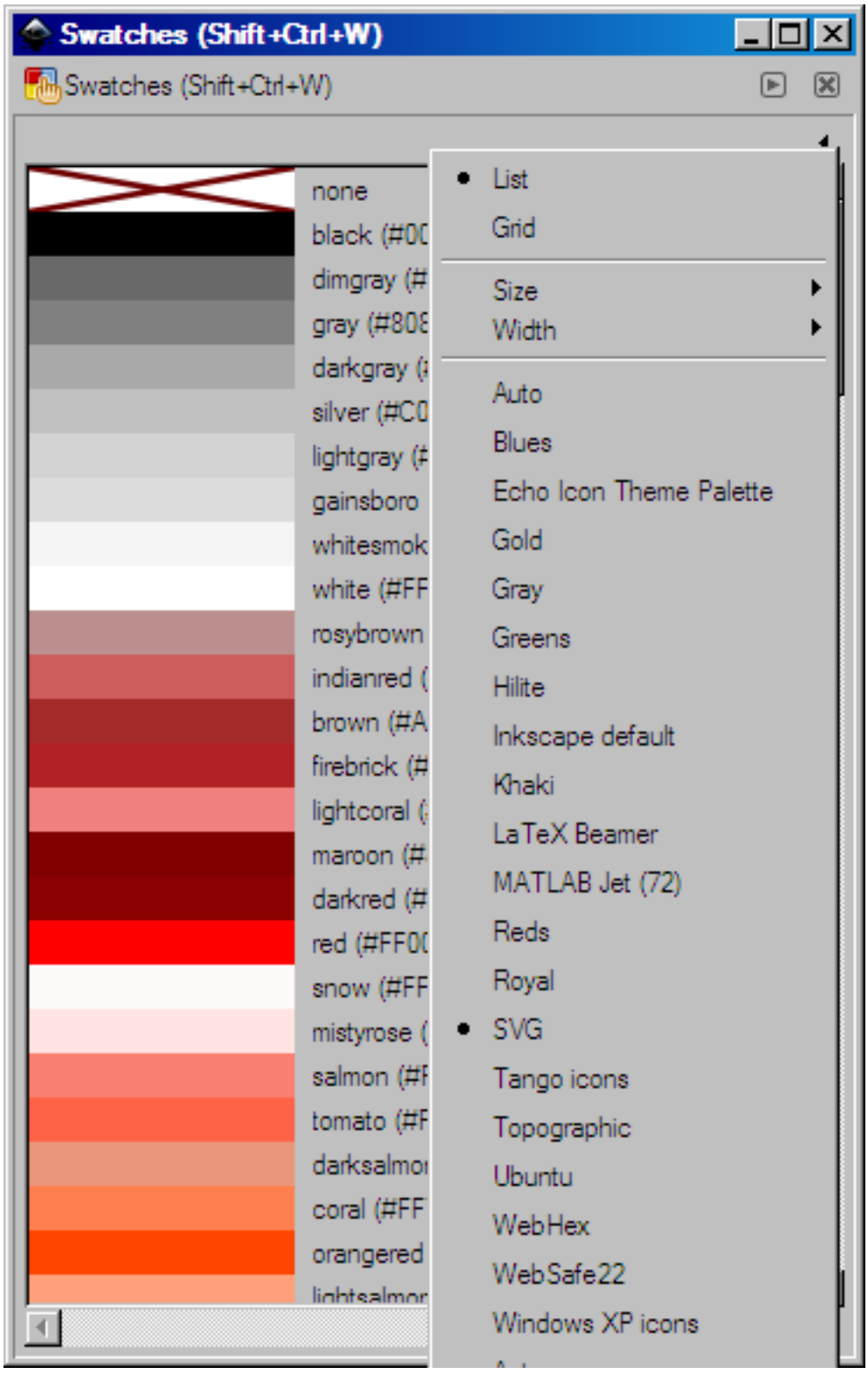
Note that the system will take the latter configuration if the live value fall between two colors configuration. For example, if 0 is red and 1 is blue, any live value between 0 and 1 will display in blue.

To get transparent blinking, enter 'none' as one of the color code. For instance, to have a red blinking effect on certain tag value, the entry shall look like 'red/none', or '#FF0000/none'. To get alternate color blinking effect, enter the desired colors accordingly, for instance, 'yellow/blue' will blink between two colors, 'white/orange/green' will blink through 3 colors in sequence.



Color Animation

If user prefer to enter color code directly, a list can be viewed by calling out Swatches pop up window by pressing Ctrl+Shift+W. Adjust the window and select 'List' and 'SVG'.

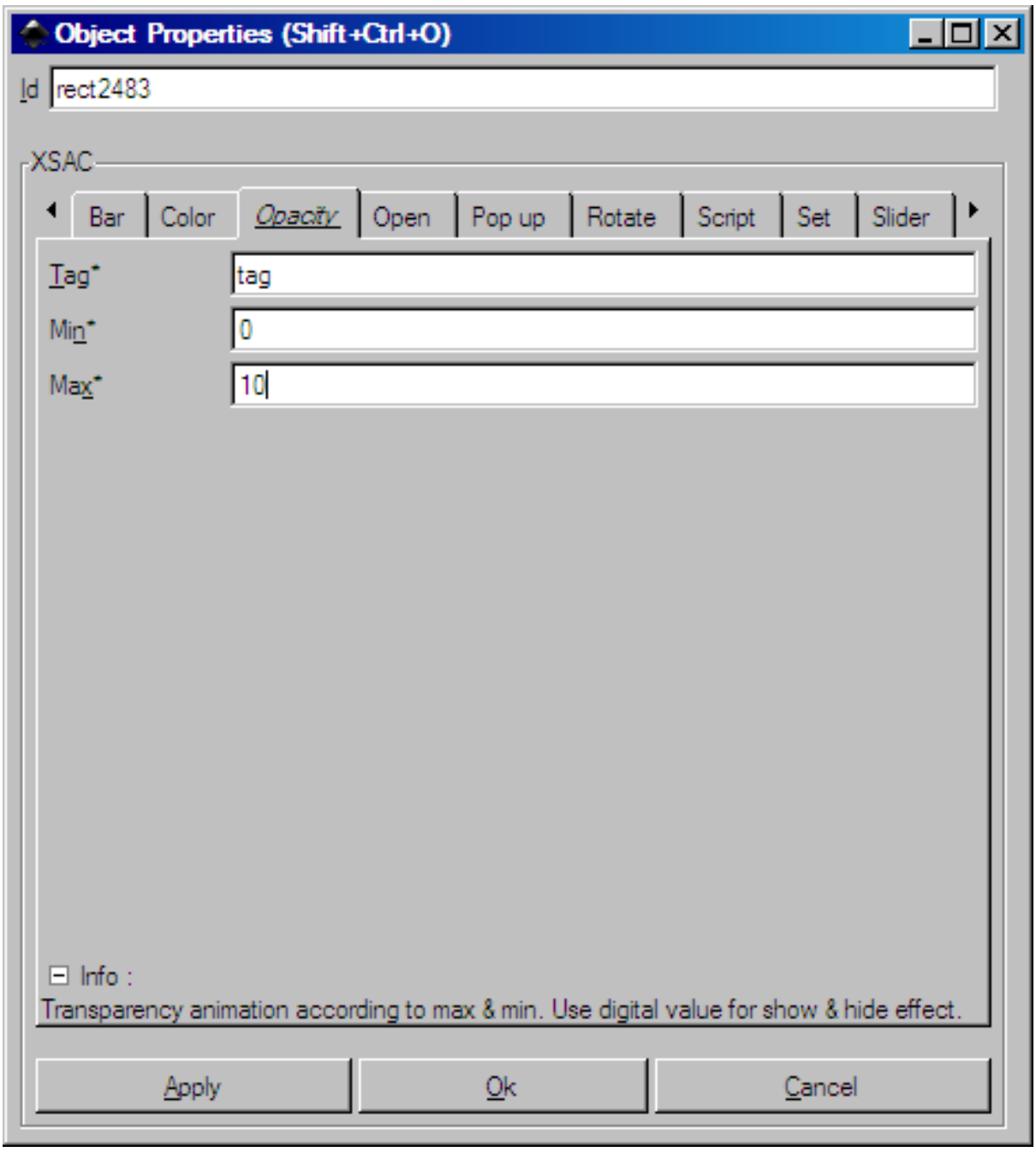


Color name listing in Swatches

For more complicated conditioned animation, logical operation can be applied just like Tag Expression by putting a leading dollar sign `$` in front of the targeted tag name. For instance, to test if a tag is fall within 1 & 10 for one specific color, the expression can be written as `1 < $tag && $tag < 10`. And then the evaluated result must be entered as `1` which is true in logical mean. Conversely, to test if a tag if fall beyond 1 & 10 for another color, separate entry of expression can be written as `$tag <= 1 || 10 <= $tag`. It's advised to always evaluate all conditions to true by entering all limit value as 1.

6.5. Opacity Animation

Opacity animation is an animation that control the transparency and solidicity of an object. There are 3 inputs: Tag, Min and Max. User will be required to enter the IntegraXor's tag to attach the event. Min and max will determine the opacity of the object, where the min will make the object disappear and max will make the object solid.



Opacity Animation

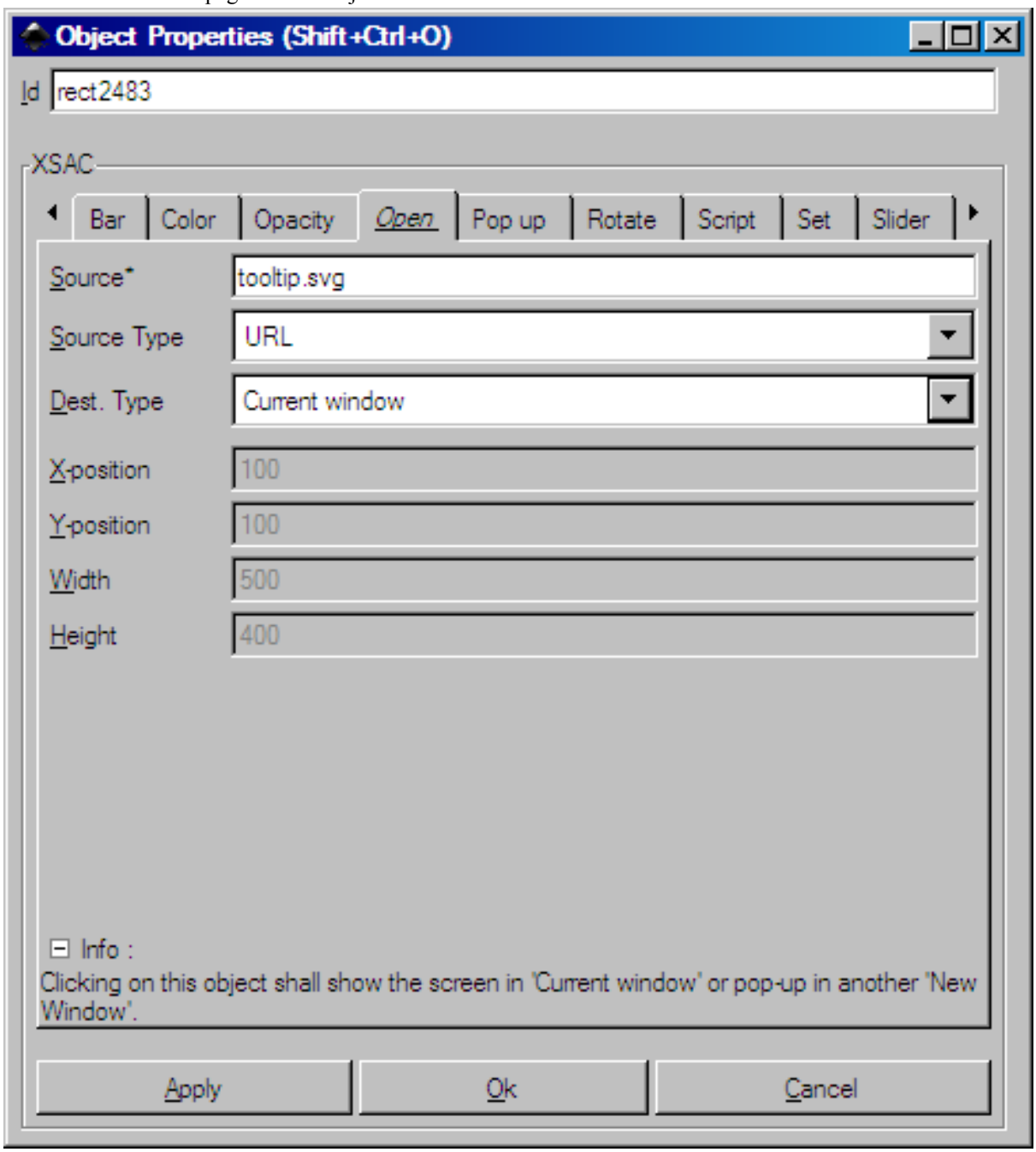
Similar to bar animation, when the value of the attached tag reached certain level, the percentage of opacity of the particular object will be change, in respective to the max and min defined. For example, if min is defined to 0 and max is 1, there will be only two type of occurance: appear or disappear.

Tip

Opacity animation can be implement in reverse effect as described. This can be done by making the maximum smaller than minimum value.

6.6. Open Animation

Open animations allow user to open a new web page by clicking on an object in the SVG graphic. User is allowed to link a web page with an object here.



Open Animation

Source: This is the source directory of the web page to be linked. This column must be a valid URL address / file directory to a web site, or a IntegraXor's tag which value is a valid URL address / file directory of a web site.

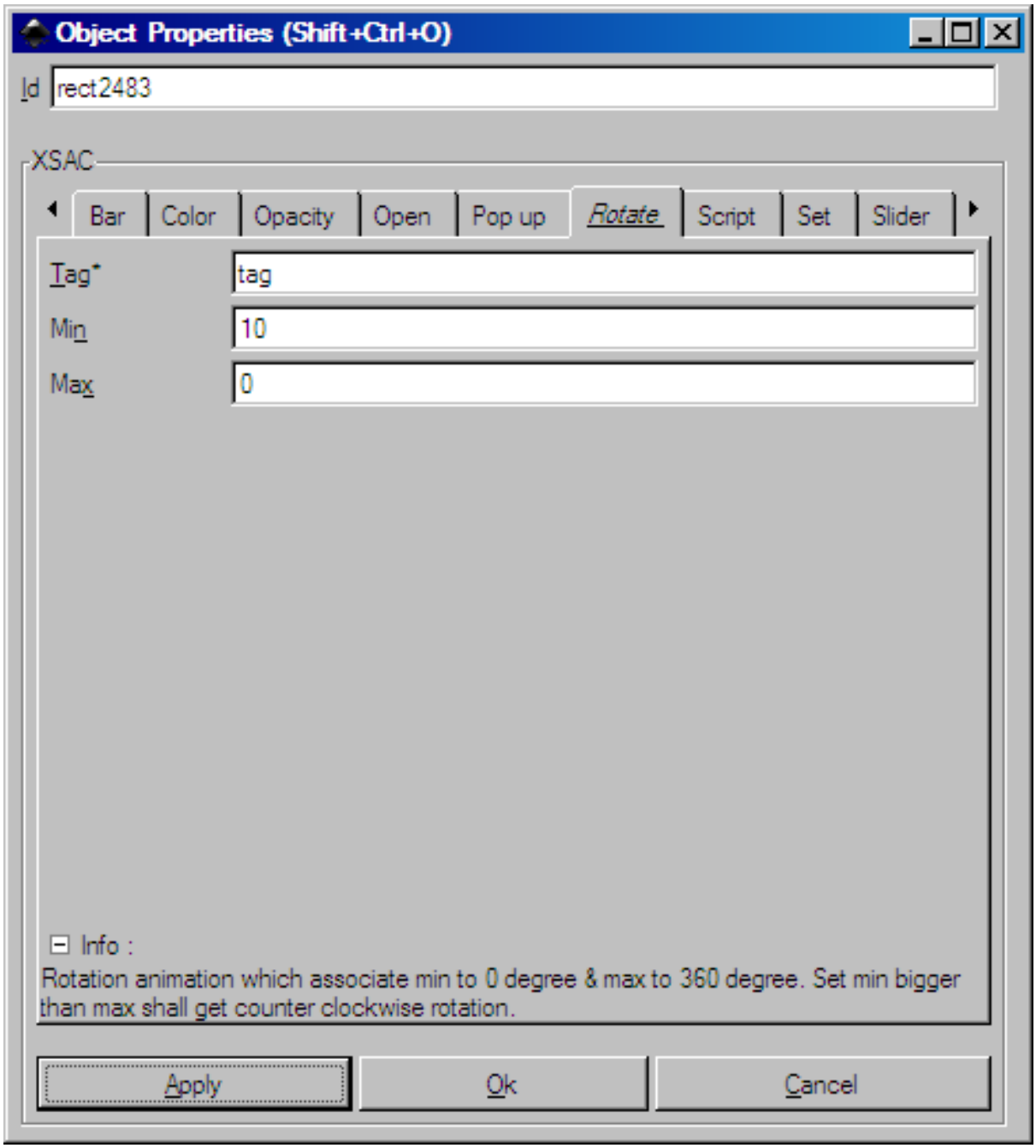
Source Type: This column determine whether the Source column is an URL (directory to web site), or it is a IntegraXor's tag. When URL is selected, Source column will be treated as a URL, and user will be redirect to the URL once the object is clicked. When TAG is selected, Source column will be treated as a IntegraXor tag, and the system will expect a valid URL in the tag value. User will be redirect to the URL inside the tag value. Therefore, user can change the destination with other factors by changing the tag value by scripts.

Dest. Type: This column determine how the new page to be opened. User is allowed to open the new page in any frame of his/her web SCADA, or open a new window. Select *New Window* will allow user to pop up a new window, while *Current Window* will open a page in the particular frame.

X-position, Y-position, width, height: These are the properties of the window if it is defined to be open in new window. X and Y position define the position of window, while width and height define the size of the window.

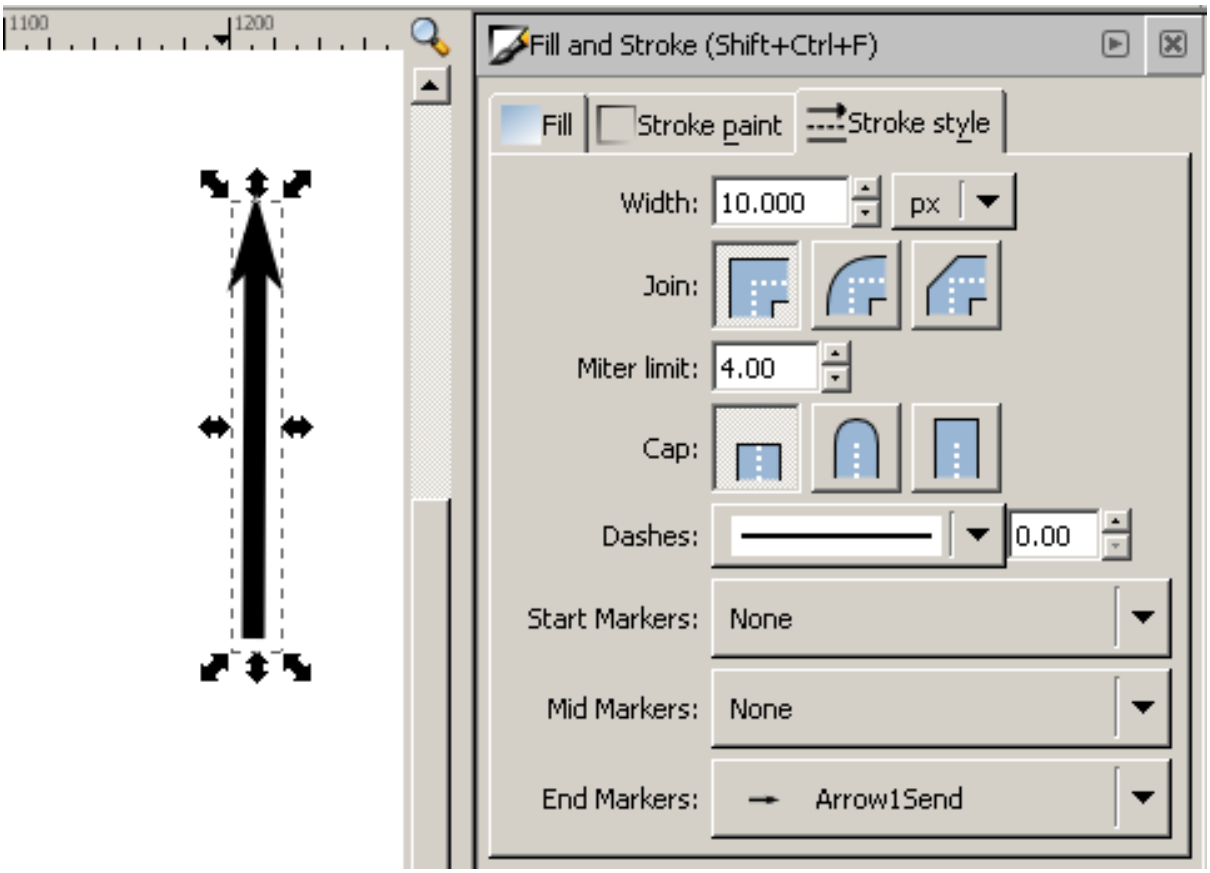
6.7. Rotate Animation

Similar to bar and opacity animation, rotate animation is an animation that allow user to rotate an object according to the percentage of maximum and minimum, in respect to the value of the tag attached, in a form of 360º degree.

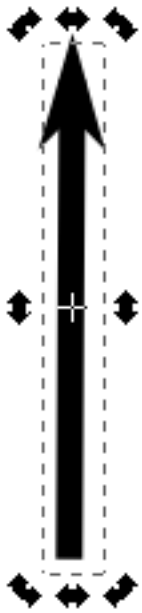


Rotate Animation

The axis of rotation is always default to the center of an object or a group of objects. To change the center of rotation, the crosshair can be moved to the desired position for this purpose. Below is a stroke with an arrow end for this illustration.

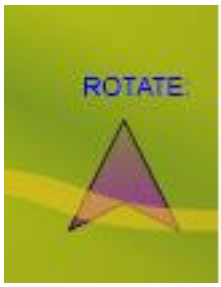


When object is first selected (first click).

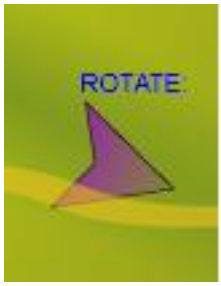


On second click, object is ready for rotation, note the crosshair (+ sign) in the middle, drag it to the desired position as the axis of rotation. It can be dragged out of object if necessary.

Example:



An object is drawed as shown, and rotate animation is configured with min 0 and max 10. When the tag value is 0, it will remain unchange.



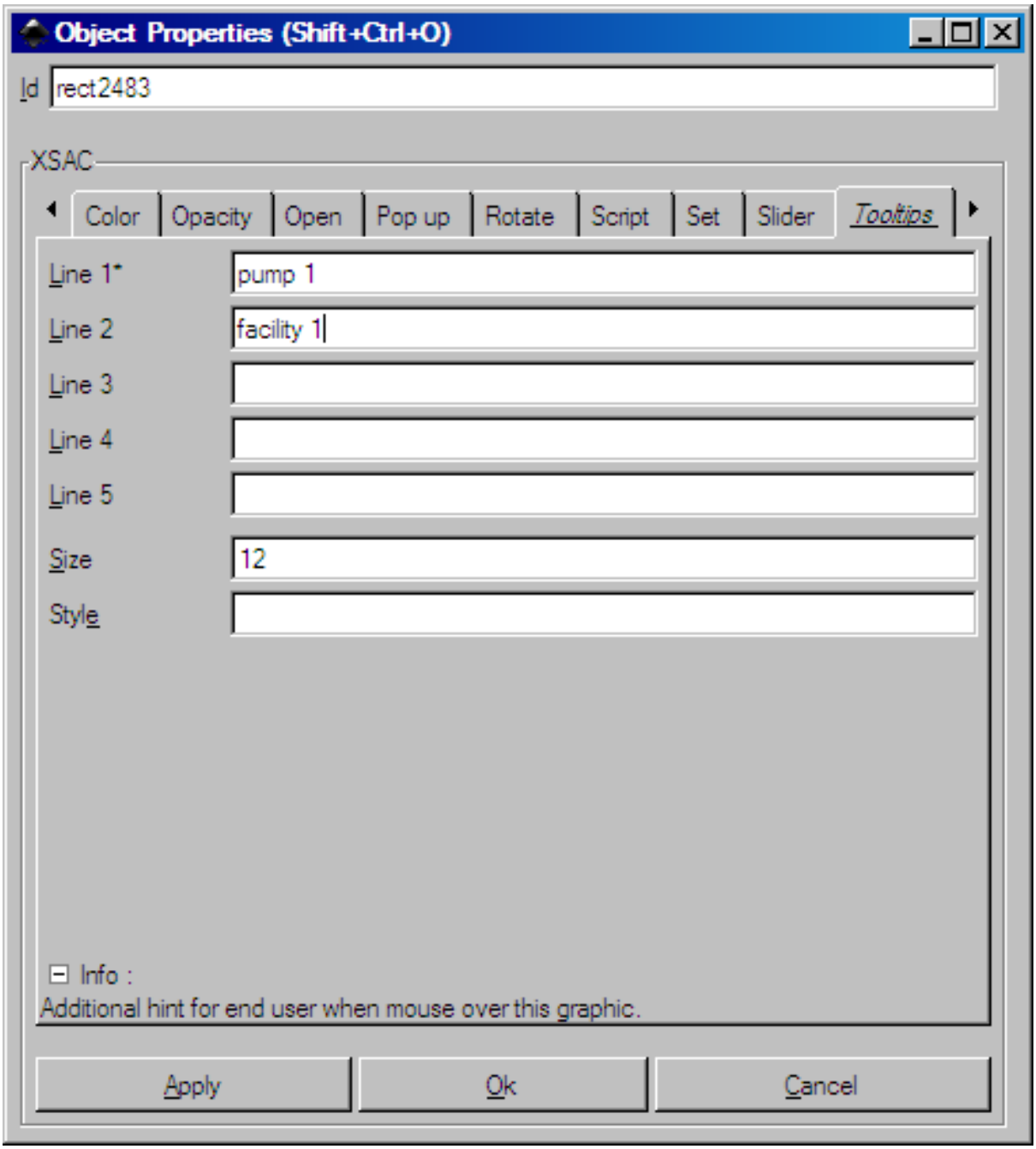
When tag value is 3, the object will rotate in an angle of 3/10 x 360, that is 108º degree.

Tip

Anti-clockwise rotation can be done by setting the minimum bigger than maximum.

6.8. Tooltip Animation

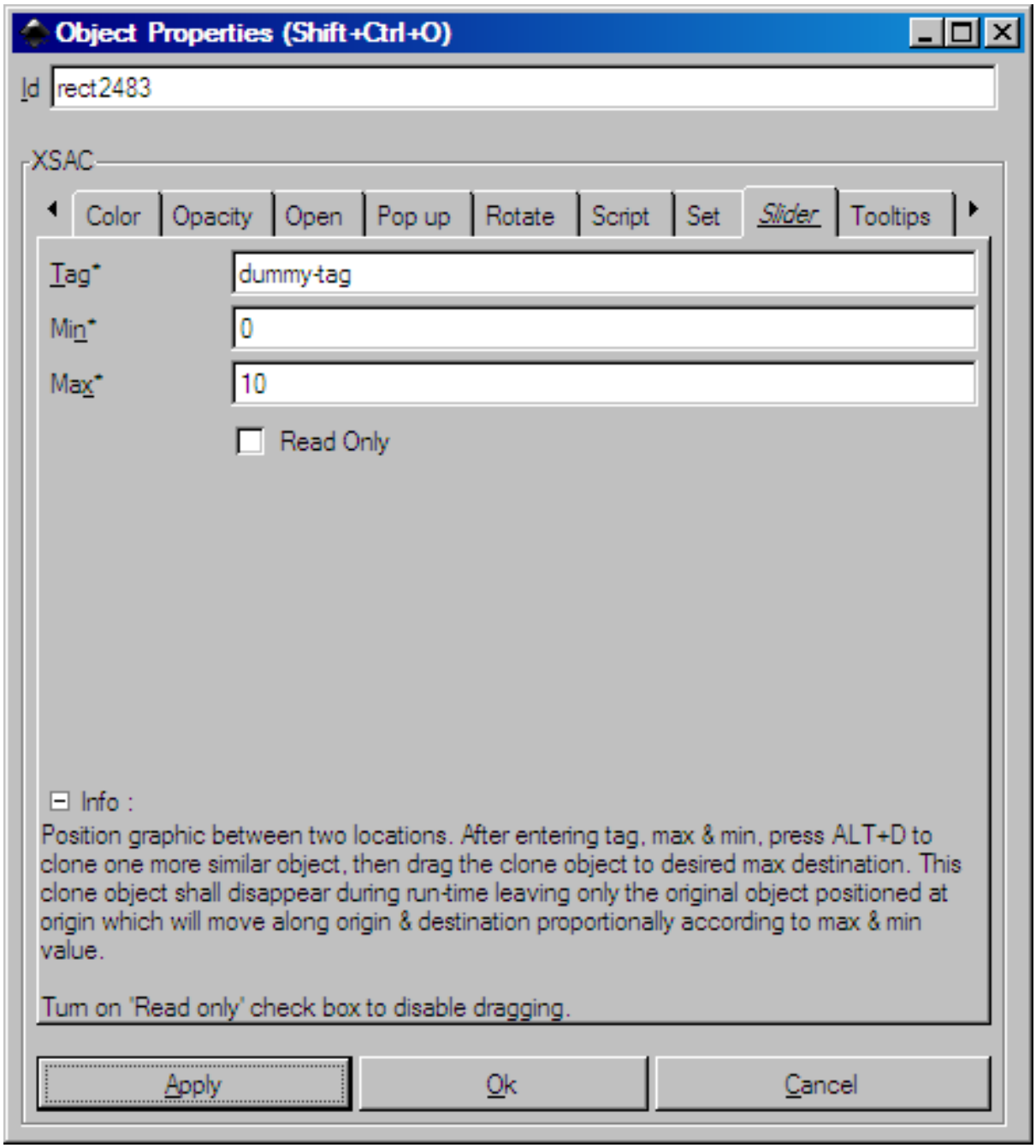
Tooltip Animation is designed to show more details to operator when mouse over an animated object. User can configure up to 5 lines which is sufficient in most cases. Text size can be specified in Size field, and advanced user can specified Style by entering CSS syntax.



Tooltip Animation

6.9. Slider Animation

Slider animation is an animation that allows user to set value for tags by dragging an object in a straight line, similar to analogue switches.



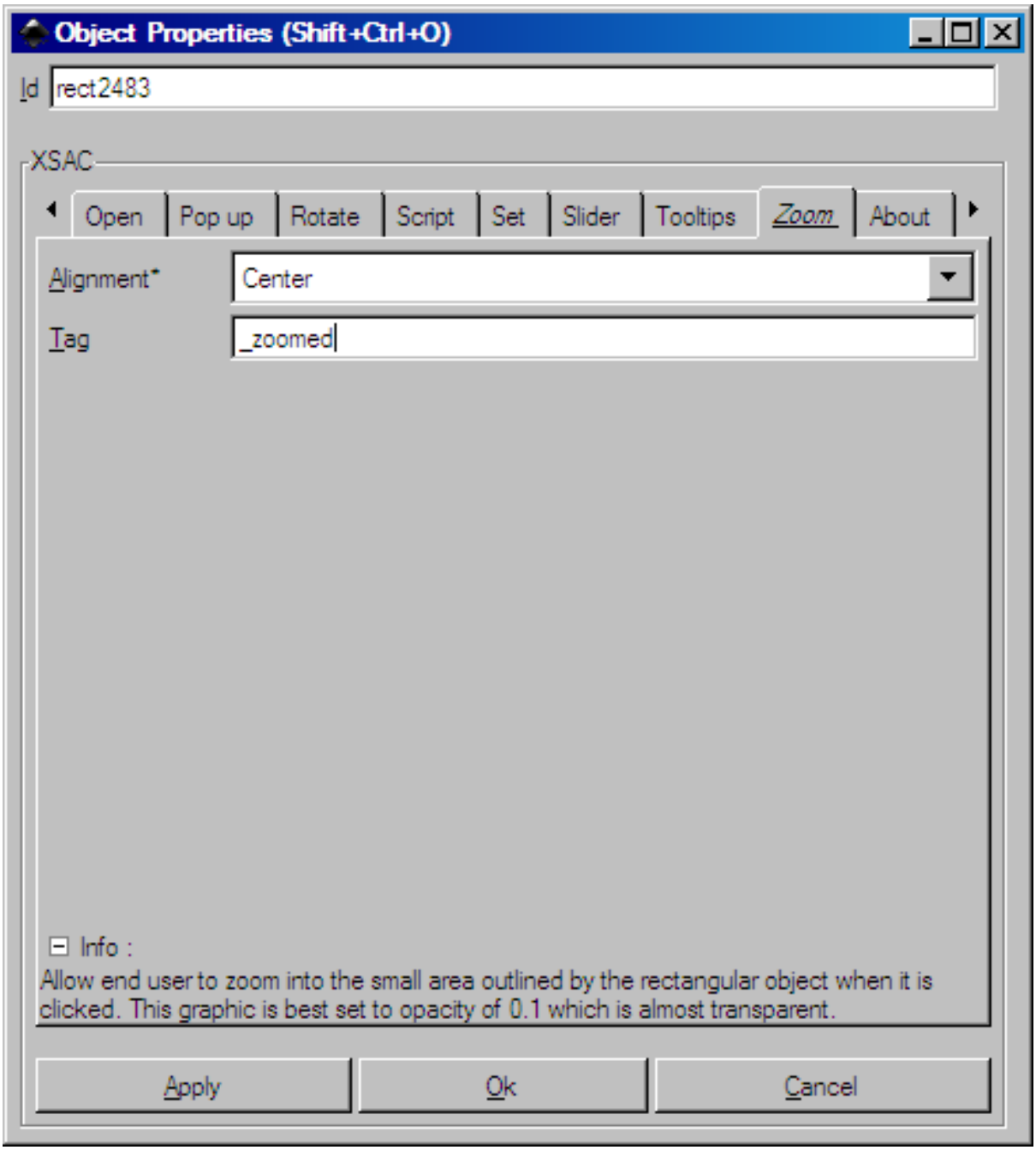
Slider Animation

To create a slider, user is required to enter the tag to be attached, and the minimum and maximum value of the tag. Check "read-only" checkbox if the slider is only for display purpose, eg, a thermometer. Then, create a clone of the object by clicking Edit > Clone > Create Clone, or simply short cut ALT+D. A clone will be created on top of the object. User is then required to move this clone to a position, where the destination of this clone object will behave as the end position of the slider, and the original object's position will be the beginning position of the slider.

During runtime, the slider object will move along its original position and the destination position in straight line, and distance between the two end will be percentage of maximum and minimum value of the tag. If "read-only" is not checked, user will be able to click and drag the object to set value into the tag.

6.10. Zoom Animation

Zoom animation allow user to zoom into specific location with single click.



Zoom Animation

Screenshot above is an example to create zoom animation. User is advised to create a low opacity rectangle object and add zoom animation as shown above. Alignment column is necessary to define where the user intends to zoom, according to the object's alignment. Tag is reserved for future use.

Important

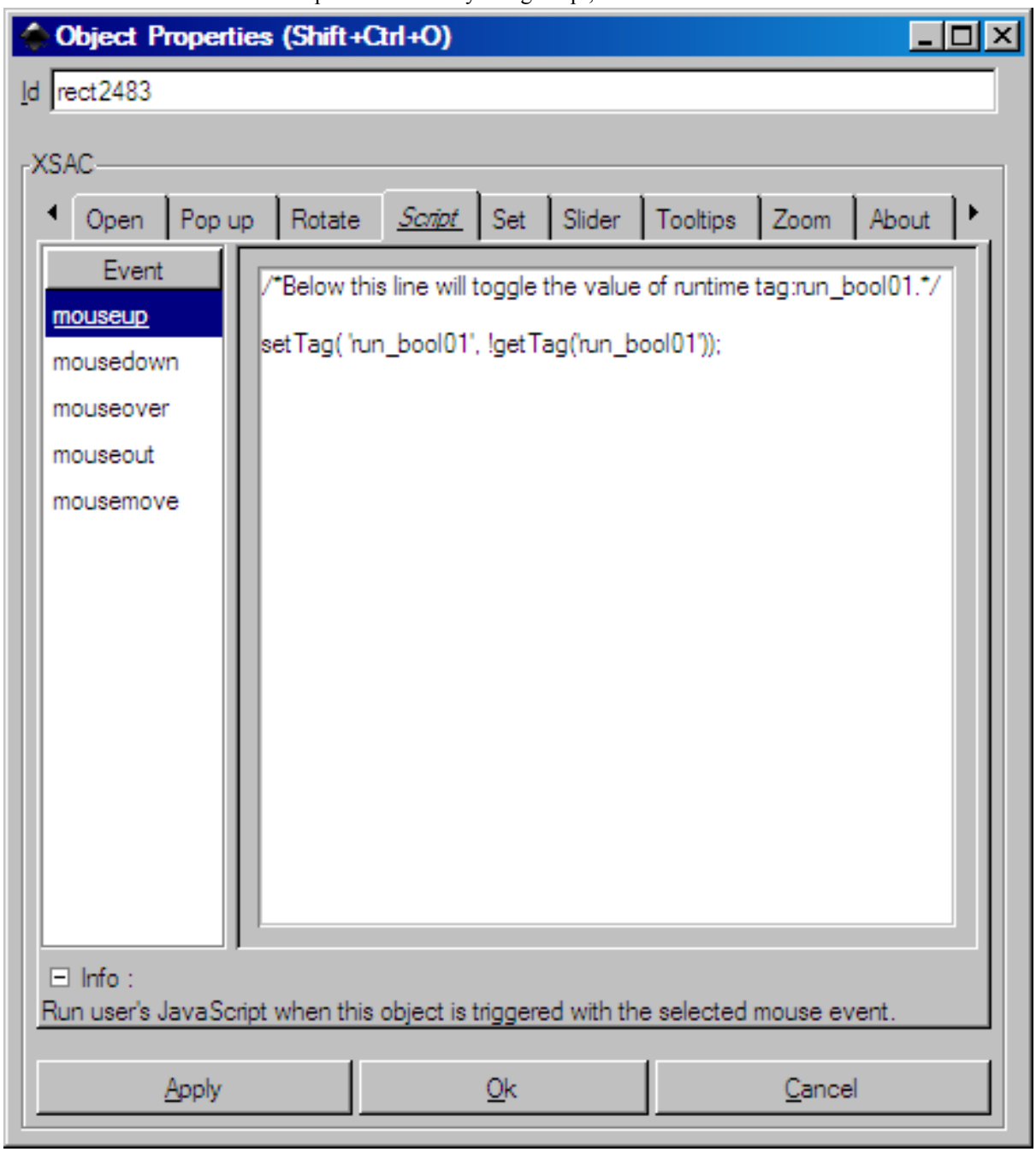
User should understand that the zoom object will block any other clickable animation behind it. If there is any clickable object behind zoom object, it will be clickable after it is zoomed.

Tip

Graphic that is zoomed-in can be zoom-out by clicking unzoom button at top right corner of the screen or by refreshing the page.

6.11. Script Animation

Script animation allow user to run a script upon click on an object. The language of script used is Javascript. User is allowed to do various complicated action by using script, such as calculation.



Script Animation

Screenshot above is an example to add script animation. This example is a simple calculation that can be done by script animation. It calculate the sum of value of tag1 and tag2, and display in a popup window.

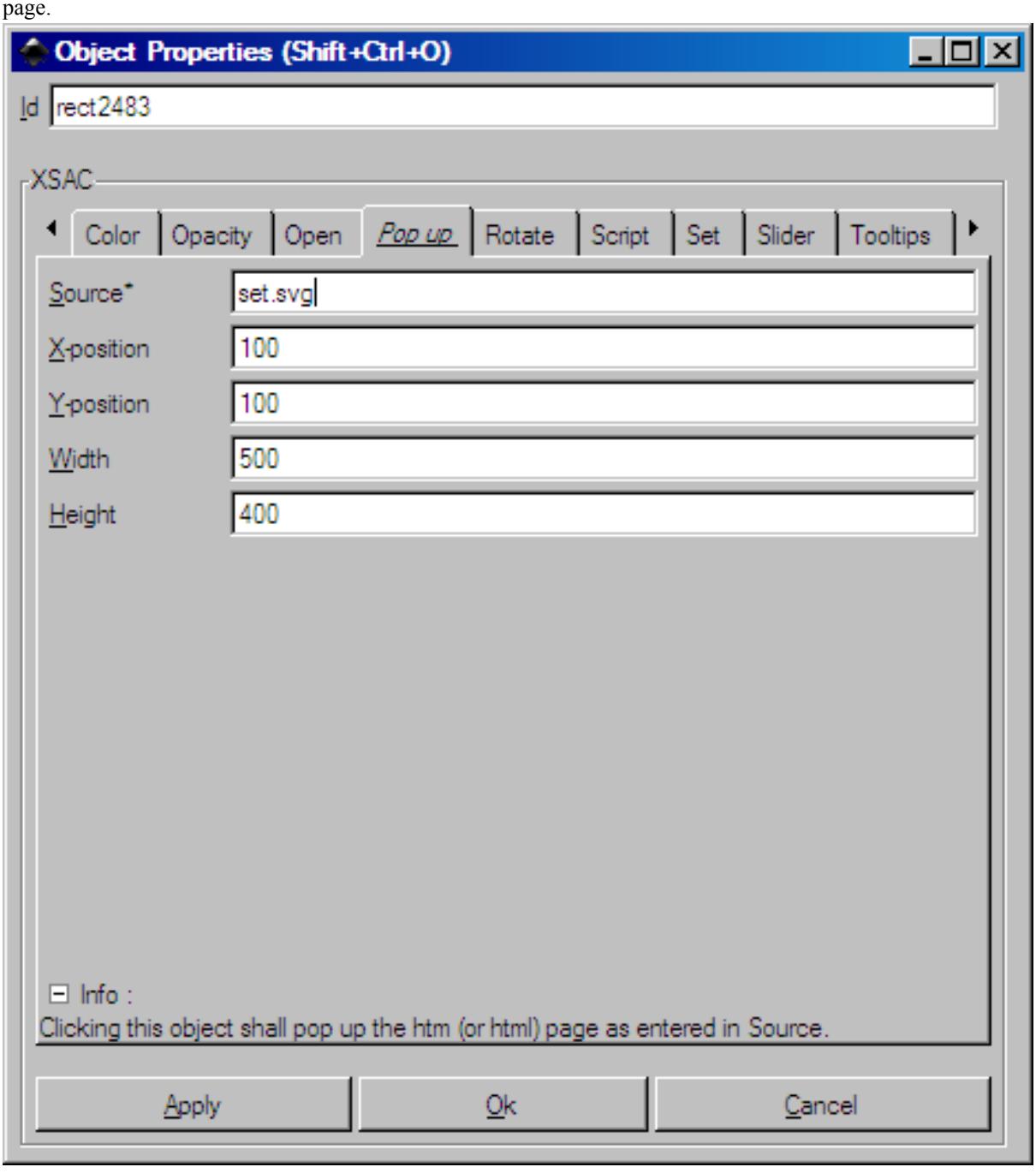
```
alert("Result: " + (getTag('tag1') + getTag('tag2')));
```

Another more practical example is to toggle a tag value, which also can be done in just one line of script. User can then further animate its status by using Color animation.

```
setTag( 'tag_bool', !getTag('tag_bool'));
```

6.12. Pop up Animation

Pop up animation allow user to pop up a window within the browser, with content of user-defined htm (or html) page.



Pop up Animation

Screenshot above is an example to add pop up animation. *Source* is a mandatory input, to refer which htm (or html) page the pop up will show. Other inputs are dimension of pop up and similar to *Open* animation as described above.

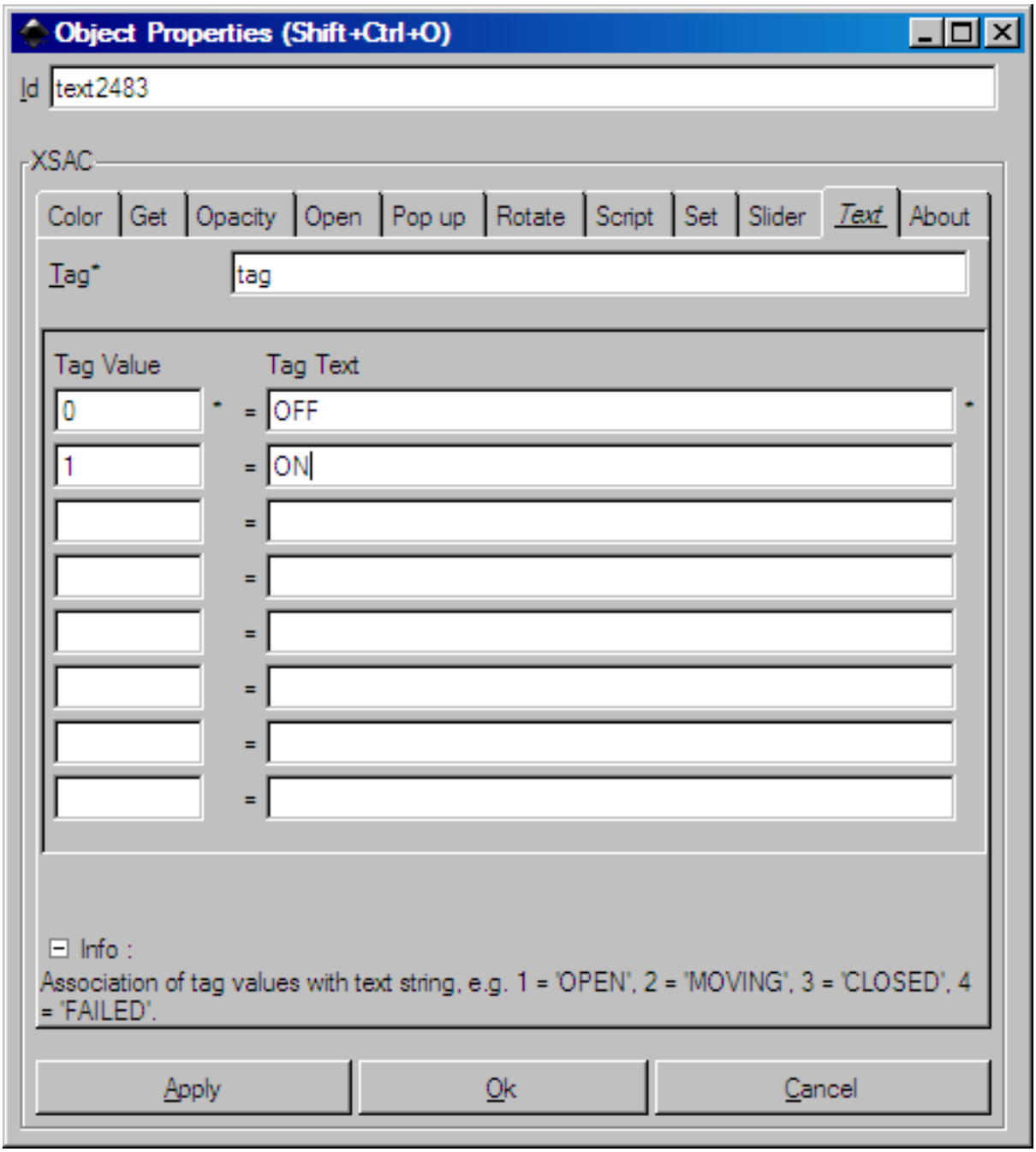
Some situations require user to pass parameters into pop up windows. User is allowed to enter parameters in *Source* column after the htm (or html) file followed by a question mark (?). For instance, the following line is valid to open up a file called 'popup.htm' and pass in a parameter called 'string' with the value of 'helloworld':

```
popup.htm?string=helloworld
```

In above example, user will receive a full URL address in popup.htm. Therefore, any information behind the question mark (?) can be further processed as received parameters by using javascript.

6.13. Text Animation

Text animation allows user to display text according to the variable with associated values. This is a convenient way to translate a field device's numerical value into human readable text. Configuring Text Animation is straightforward, simply enter the desired tag, and at least configure one set of condition for text association accordingly.



Text Animation

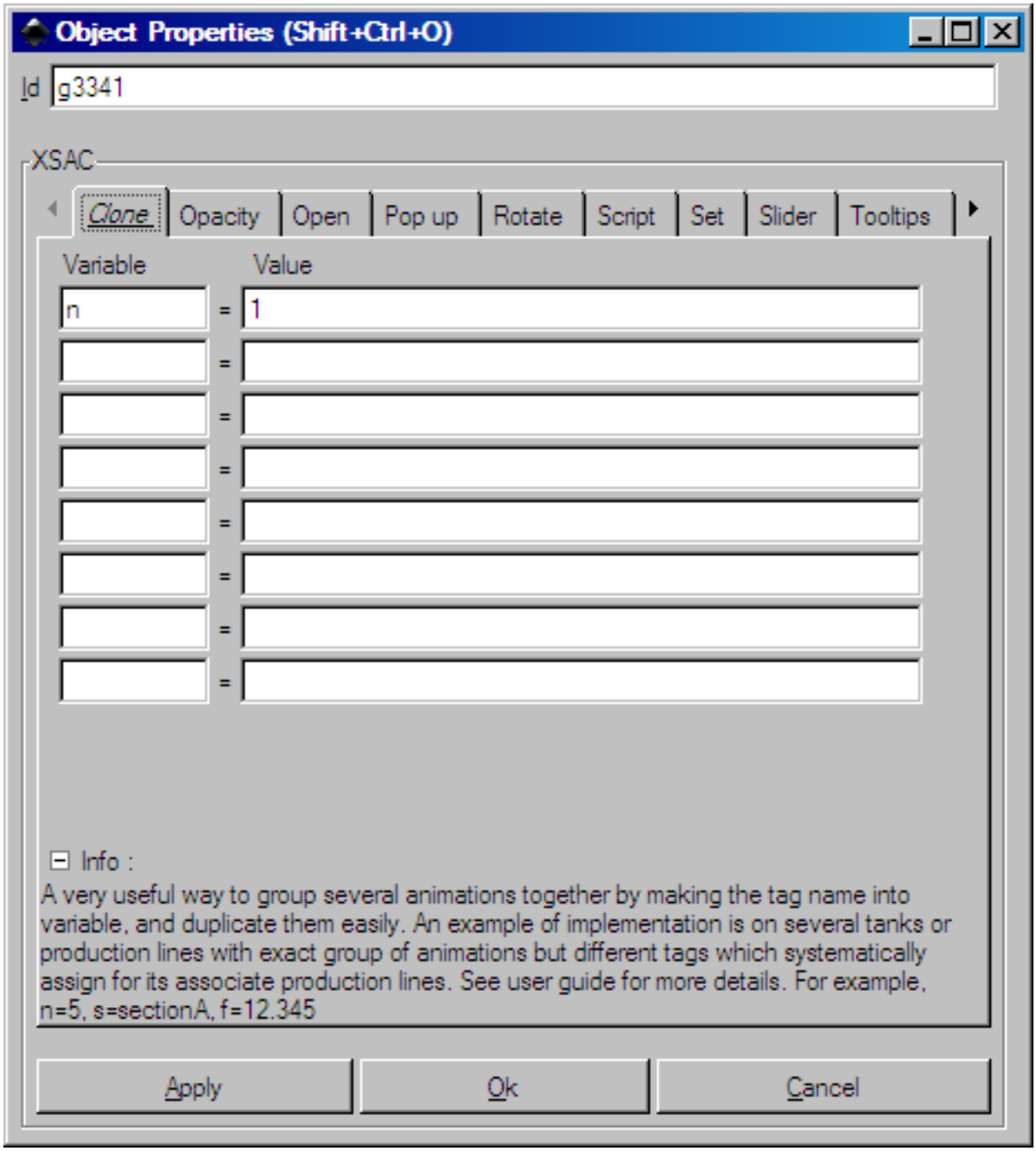
Note

Similar to Get animation, the graphic Object must be a Text to enable Text Animation tab properties. This mean rectangle or circle will not have Text Animation Tab shown in Object Properties.

6.14. Clone Animation

Clone animation allows user to duplicate a grouped object while keeping unique tag name. This is useful when there are multiple similar objects such as pumps, valves, etc, and each of the objects has same animation but associated to different tags. Therefore Clone animation interface only appear on grouped graphic object when calling out Object Properties. To use clone animation, user must first design or differentiate the tag pattern of the tags inside the graphic group.

For example, a pump has 2 animations, and user wants to create 2 pumps, lets say the tags are: `pump_1_state, pump_1_reading, pump_2_state & pump_2_reading`. The difference between pumps are the numbers 1 and 2. When user assigns the animation, enter the tag `pump_%n_state and pump_%n_reading` instead. After the pump is completed in drawing, group the drawing by CTRL+G. This should be a reuseable pump object. Now, user can duplicate this pump object by pressing CTRL+D. Each of the duplicated clone object can be assigned clone animation into it, while Variable should be `n` and Value should be `1` for the pump which uses `pump_1_state & pump_1_reading` tags in animation. For second pump, assign clone animation into the pump and make Value to be `2`.

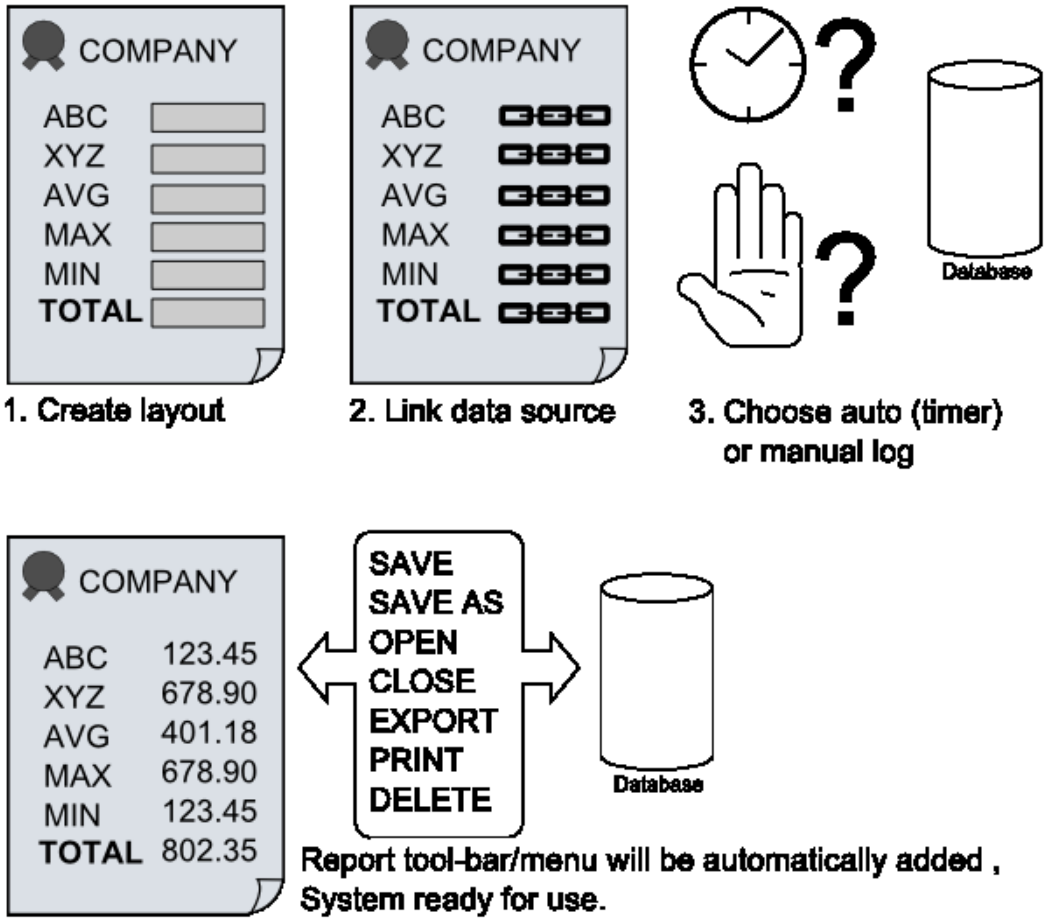


Clone Animation

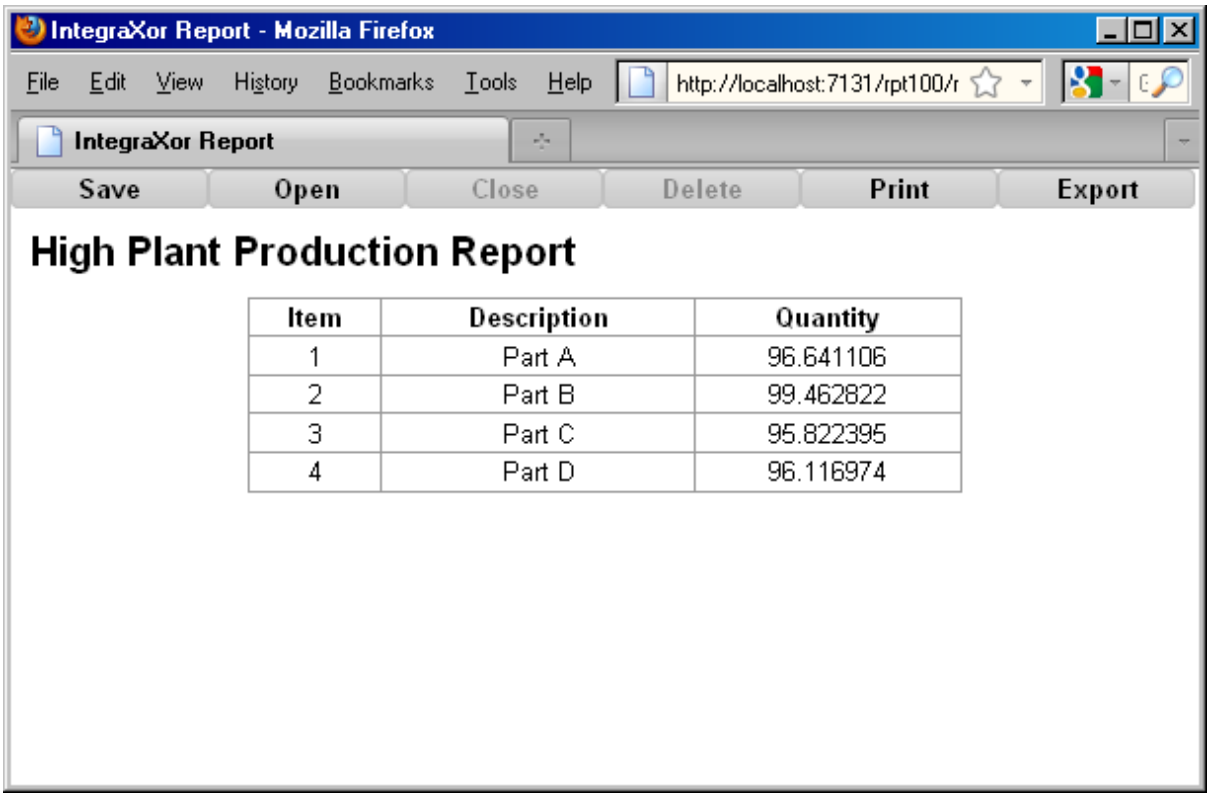
The symbol % is an identifier to the animation system, therefore it is not required to be entered in Variable field. And several variables can be entered in separate rows for use within a cloned graphic object. For example, `tag_%n_%s_%d` would have at least 3 entries in the list.

Chapter 7. Report

IntegraXor reporting module is an innovative solution which provides one time configuration for both screen viewing and hardcopy printout. The system do not require any external plug-in or rely on external program to produce a printout. Engineer only needs to produce a layout in HTML as usual, link to datasource by adding in tagname into the layout, then specify the triggering timer according to the nature of the report. For instance, hourly report shall be triggered/generated by an hourly timer, daily report shall be triggered/generated by a daily timer on user specific hour.



IntegraXor 3-steps report creation


IntegraXor Report - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:7131/rpt100/r
IntegraXor Report
Save Open Close Delete Print Export
High Plant Production Report
Item Description Quantity
1 Part A 96.641106
2 Part B 99.462822
3 Part C 95.822395
4 Part D 96.116974

A typical snapshot report collect all data in an instant of time

7.1. Snapshot Report

Layout Creation

First essential step to produce a report is to create a report layout. Ensure the HTML report layout file is XML compliant, in another word, it's XHTML compliant. All HTML element must be written in lowercase and closed accordingly. For instance, <td> must be followed by </td> after typing its content, rather than leave it open and start by another <td>. A line break shall looks like <br/> and horizontal break shall be <hr/>.

Tip

Engineer who's beginner in HTML editing may prefers visual editing tool like KompoZer.

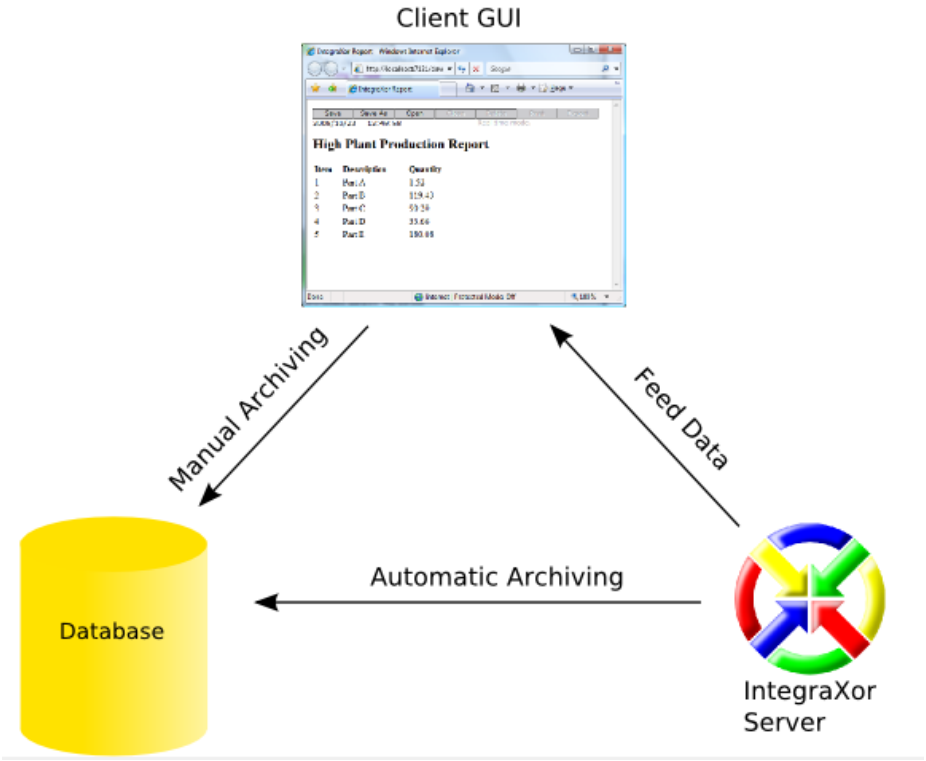
Data Linking

Second step of linking process involved adding tags into layout, and adding linkage to report module.

Server Archiving

Final step is on server side configuration, which involving calculating data and specifying the timer for archiving. It is S.I. engineer's role to make the data available in server memory, whether to poll the data from external source or a result of calculations.

Archiving or saving into disk is a commonly needed feature. It can be done manually by end user on GUI during runtime, or automatically archived/generated based on preset timer.



Manual and Automated Report Archiving Architecture

7.1.1. Creating Layout

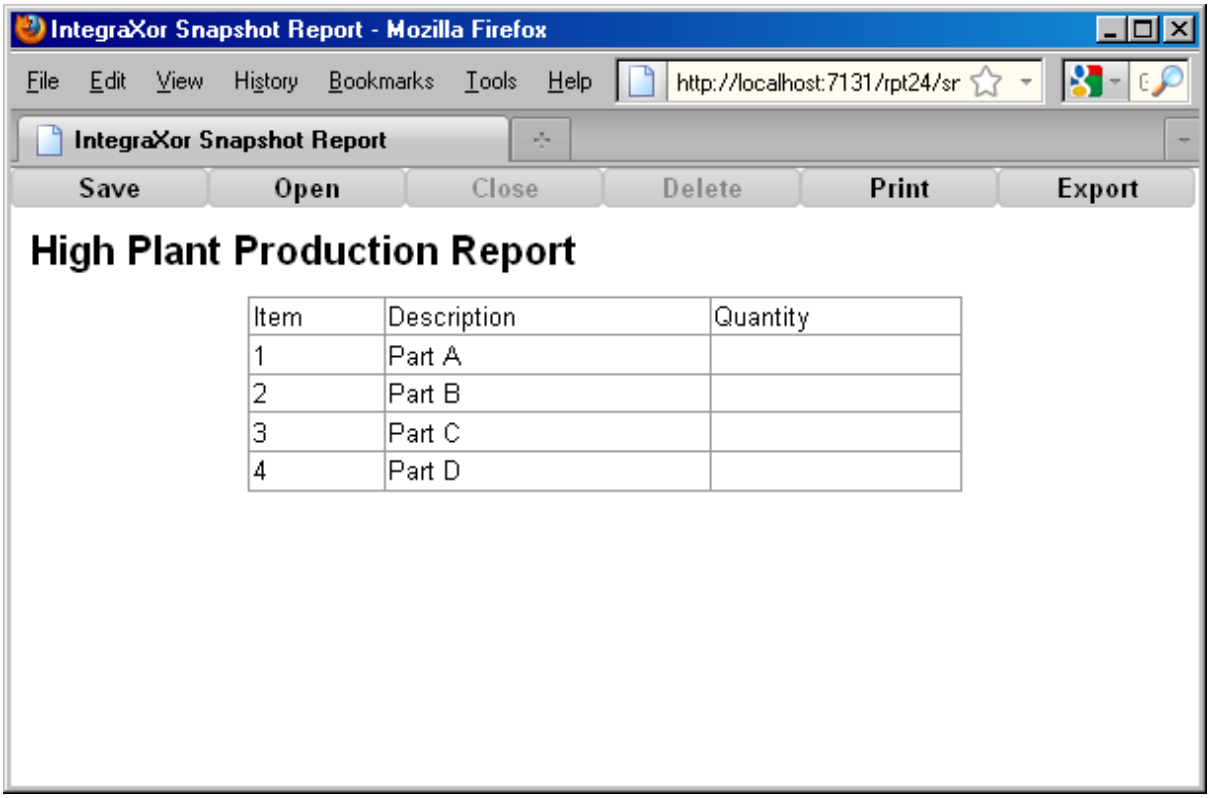
Below is a sample of standard compliant HTML layout for a production report. Note the highlighted script inclusion is needed to add-in the reporting tool-bar; i.e. GUI.

```
<!DOCTYPE HTML>
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>IntegraXor Snapshot Report</title>
        <link rel="stylesheet" type="text/css"
                href="system/styles/igrX.css" />
    </head>
    <body>
        <div id="idButtons" class="paragraph"></div>
        <h1>High Plant Production Report</h1>
        <table width="60%">
            <tr>
                <th>Item</th><th>Description</th><th>Quantity</th>
            </tr>
            <tr>
                <td>1</td>
                <td>Part A</td>
                <td></td>
            </tr>
            <tr>
                <td>2</td>
                <td>Part B</td>
                <td></td>
            </tr>
            <tr>
                <td>3</td>
                <td>Part C</td>
                <td></td>
```

```
            </tr>
            <tr>
                <td>4</td>
                <td>Part D</td>
                <td></td>
            </tr>
        </table>
    <script type="text/javascript" src="./system/scripts/igrX.js"></script>
    <script type="text/javascript">
        var rpt = igrx.report();
    </script>
    </body>
</html>
```

Tip

Engineer who's beginner in HTML editing may prefers visual editing tool like KompoZer.



Output in browser from above listing

The output from the above listing can be seen once the layout is added into the mimic menu.

7.1.2. Link to Datasource

Once a layout is created, engineer needs to associate a dynamic field with the desired tag. To associate a data field with a tag, add in a JSON compliant syntax as its data attribute like this: `data-igrx="{attr:'get',tag:'tagname'}"`. Below highlighted section is the sample after adding in the tag links.

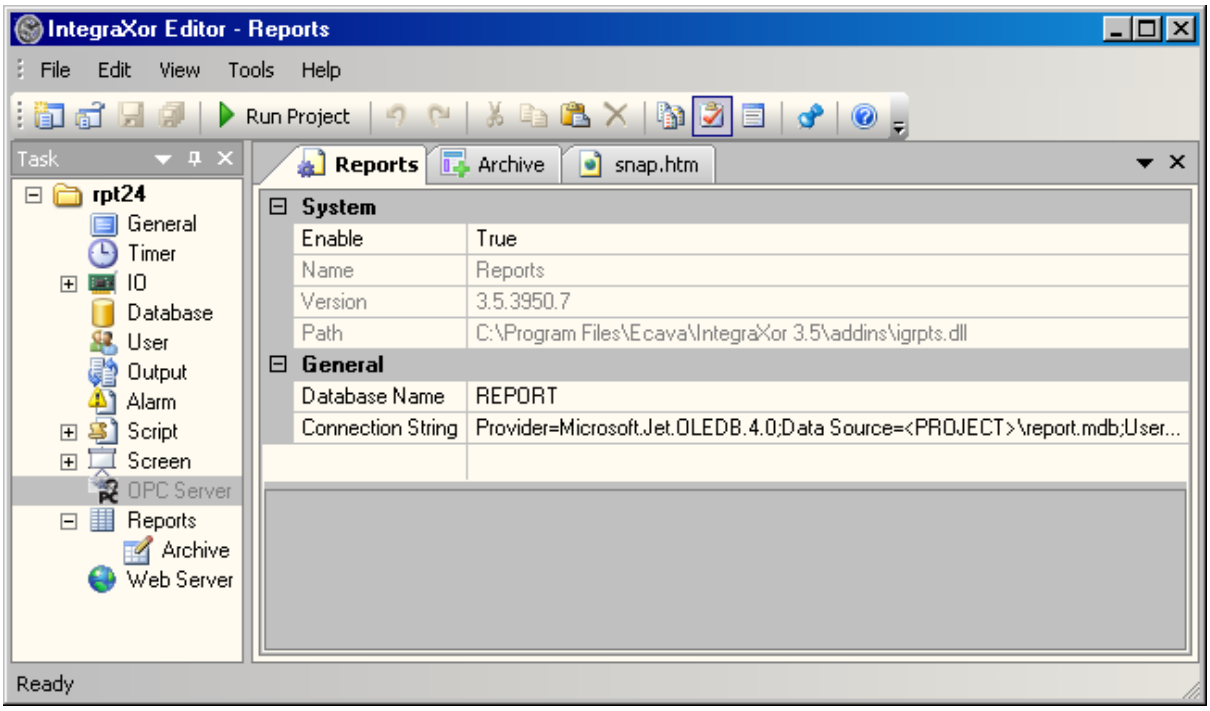
```
<!DOCTYPE HTML>
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
```

```
        <title>IntegraXor Report</title>
        <link rel="stylesheet" type="text/css"
              href="system/styles/igrX.css" />
    </head>
    <body>
        <div id="idButtons" class="paragraph"></div>
        <h1>High Plant Production Report</h1>
        <table width="60%">
            <tr>
                <th>Item</th><th>Description</th><th>Quantity</th>
            </tr>
            <tr>
                <td>1</td>
                <td>Part A</td>
                <td data-igrx="{attr:'get',tag:'part_a'}"></td>
            </tr>
            <tr>
                <td>2</td>
                <td>Part B</td>
                <td data-igrx="{attr:'get',tag:'part_b'}"></td>
            </tr>
            <tr>
                <td>3</td>
                <td>Part C</td>
                <td data-igrx="{attr:'get',tag:'part_c'}"></td>
            </tr>
            <tr>
                <td>4</td>
                <td>Part D</td>
                <td data-igrx="{attr:'get',tag:'part_d'}"></td>
            </tr>
        </table>
    <script type="text/javascript" src="./system/scripts/igrX.js"></script>
    <script type="text/javascript">
        var rpt = igrx.report();
    </script>
    </body>
</html>
```

7.1.3. Archiving Report

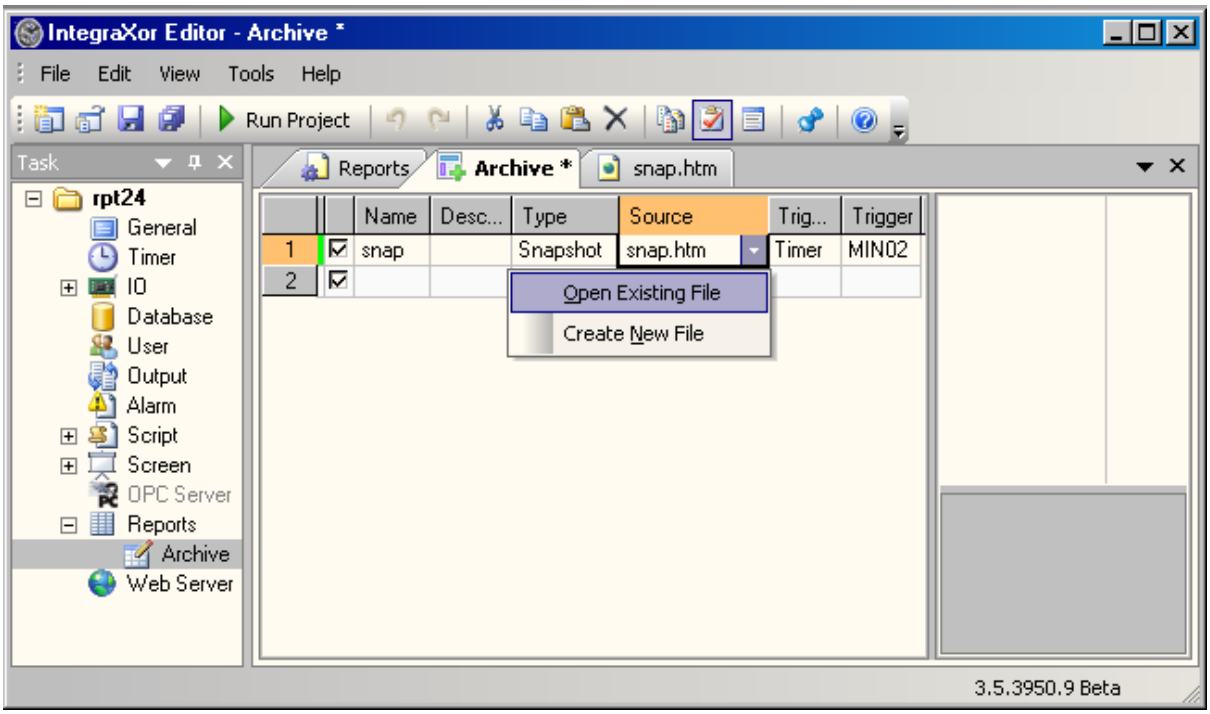
The third and final step is to specify the time of report generation. Assuming this sample is to generate a daily production report at 6am. Then all data shall be calculated by 6am for use. If there's a server script for this process, then it shall be triggered by a timer at 6am. In this sample, the tags for 'part_a', 'part_b', 'part_c', 'part_d' and 'part_e' shall have its data furnished by 6am.

Auto archiving shall be done with slight delay after the calculation triggering time. Depending on the external factor like communication delay or time synchronization problem between PC and devices, the optimum delay can varies from few seconds to 5 minutes, or even more for extreme scenario. In this sample we assume the data would be ready in the most 5 seconds, so we can create a timer at 06:00:05, and trigger the report above daily.



Database configuration for report

General configuration allows user to customize report database name as well as the connection string to database. Default workable entry will save report archive into local Access report.mdb file.



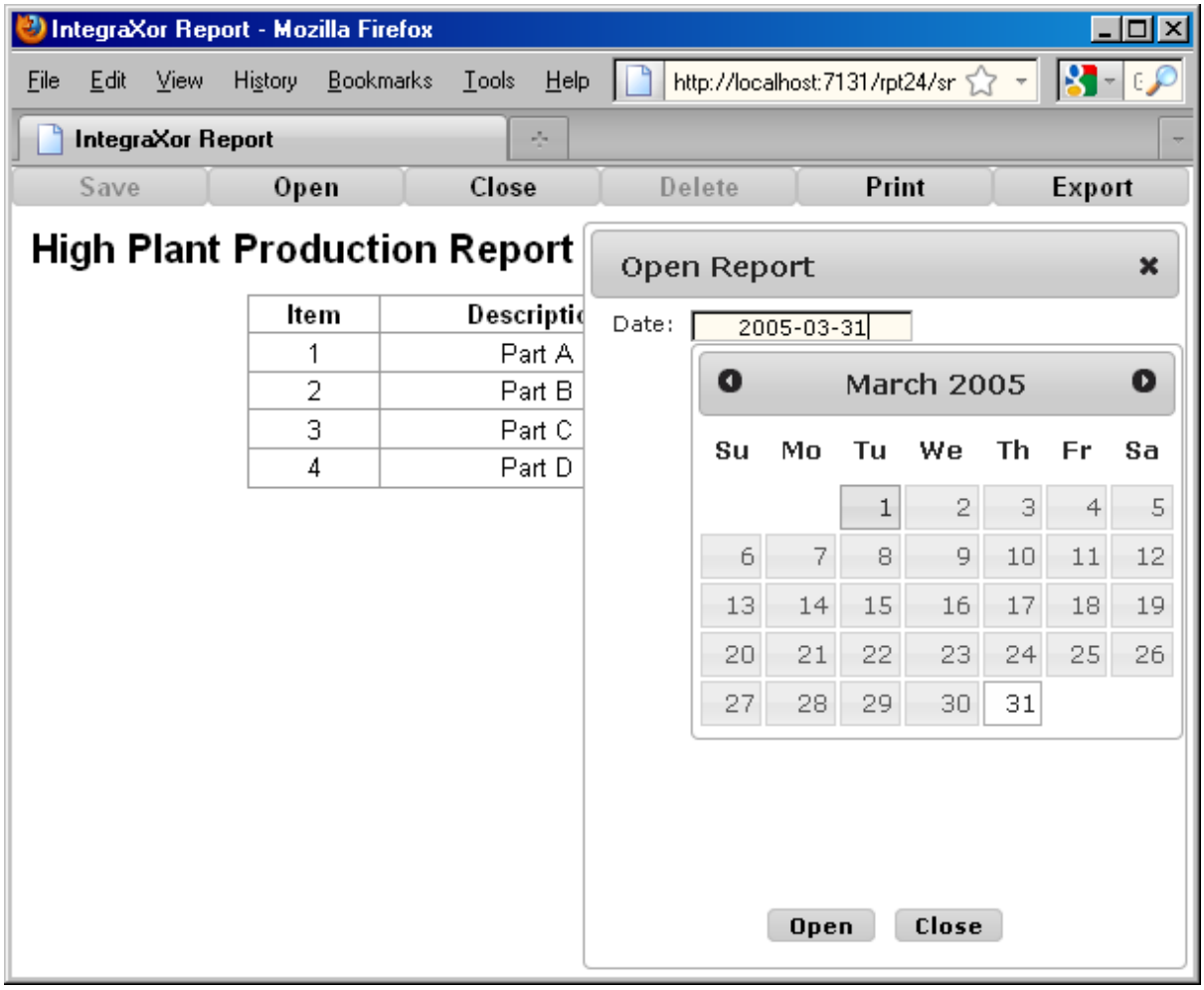
Report archiving configuration

At task panel on the left, expand Report and click on Archive. List of reports will be opened.

Field	Description
Name	Name of the report. Alphabets and numbers only.
Description	Description of the report

Field	Description
Type	Snapshot or Statement report
Source	HTML source file to archive.

All the inputs are mandatory except Description. Report type is default to *snapshot*, which means report server will capture tag values and store into database after report generation is triggered either by tag or timer. Batch event can be configured at right property panel after the type of the report is selected.



Opening report archive

Note

For any project that using existing or previously created database, be aware that text entered in *Name* will be used as the table of report database. Engineer should avoid using duplicate table name if existing database already contains other tables.

7.2. Statement Report

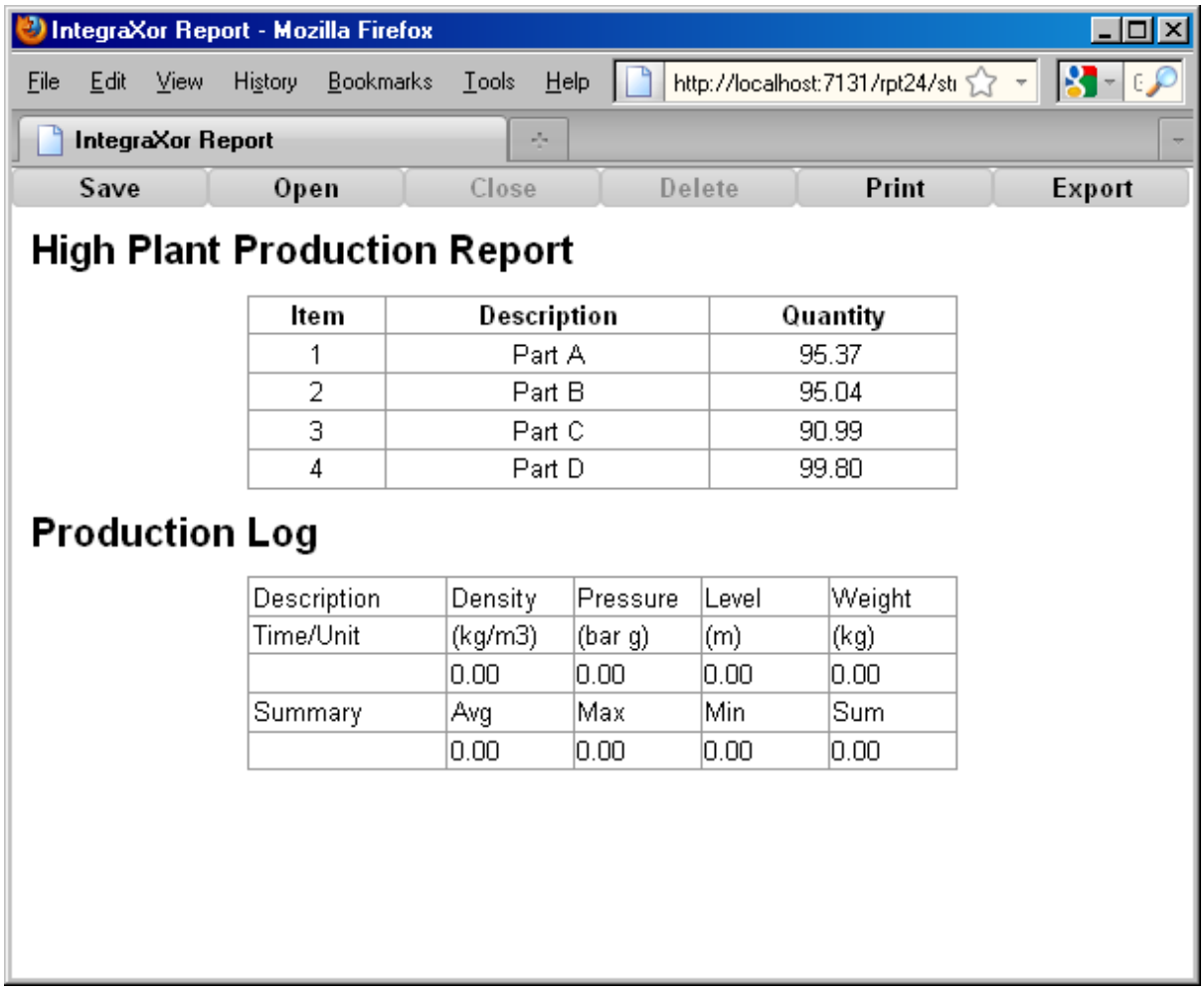
This powerful report module allows data accumulation in a growing table as a statement report. Common example of statement report is like phone bill that have call logging, banking statement with transaction listing, credit card transaction statement and etc. However, for industrial automation, statement report normally needed in batching process that span through a certain duration of production time.

7.2.1. Layout Configuration

To add accumulation data insertion into a report, the syntax must be entered into a table, so that the data will grow within the table row. Below showing highlighted section of table to cater for this purpose.

```
<!DOCTYPE HTML>
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>IntegraXor Report</title>
        <link rel="stylesheet" type="text/css"
              href="system/styles/igrX.css" />
    </head>
    <body>
        <div id="idButtons" class="paragraph"></div>
        <h1>High Plant Production Report</h1>

        <table width="60%">
            <tr>
                <th>Item</th><th>Description</th><th>Quantity</th>
            </tr>
            <tr>
                <td>1</td>
                <td>Part A</td>
                <td data-igrx="{attr:'get',tag:'part_a'}">0.00</td>
            </tr>
            <tr>
                <td>2</td>
                <td>Part B</td>
                <td data-igrx="{attr:'get',tag:'part_b'}">0.00</td>
            </tr>
            <tr>
                <td>3</td>
                <td>Part C</td>
                <td data-igrx="{attr:'get',tag:'part_c'}">0.00</td>
            </tr>
            <tr>
                <td>4</td>
                <td>Part D</td>
                <td data-igrx="{attr:'get',tag:'part_d'}">0.00</td>
            </tr>
        </table>
        <h1>Production Log</h1>
        <table width="60%">
            <tr>
                <th width="28%">Description</th>
                <th width="18%">Density</th>
                <th width="18%">Pressure</th>
                <th width="18%">Level</th>
                <th width="18%">Weight</th>
            </tr>
            <tr>
                <td>Time/Unit</td>
                <td>(kg/m3)</td>
                <td>(bar g)</td>
                <td>(m)</td>
                <td>(kg)</td>
            </tr>
            <tr>
                <td></td>
                <td>0.00</td>
                <td>0.00</td>
                <td>0.00</td>
                <td>0.00</td>
            </tr>
            <tr>
                <td>Summary</td>
                <td>Avg</td>
```

```
                <td>Max</td>
                <td>Min</td>
                <td>Sum</td>
            </tr>
            <tr>
                <td></td>
                <td>0.00</td>
                <td>0.00</td>
                <td>0.00</td>
                <td>0.00</td>
            </tr>
        </table>
    <script type="text/javascript" src="./system/scripts/igrX.js"></script>
    <script type="text/javascript">
        var rpt = igrx.report();
    </script>
    </body>
</html>
```


IntegraXor Report - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:7131/rpt24/sti
IntegraXor Report
Save Open Close Delete Print Export
High Plant Production Report
Item Description Quantity
1 Part A 95.37
2 Part B 95.04
3 Part C 90.99
4 Part D 99.80
Production Log
Description Density Pressure Level Weight
Time/Unit (kg/m3) (bar g) (m) (kg)
0.00 0.00 0.00 0.00
Summary Avg Max Min Sum
0.00 0.00 0.00 0.00

Output in browser from above listing

7.2.2. Link to Datasource

Once a table is created, engineer needs to associate a dynamic field with the desired tag. To associate a data field with a tag for growing table row entry, add in syntax as its attribute like this: `data-igrx="{attr:'log',tag:'tagname'}"`. Below is the sample after adding in the tag links.

```
<!DOCTYPE HTML>
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>IntegraXor Report</title>
        <link rel="stylesheet"
            type="text/css" href="system/styles/igrX.css" />
    </head>
    <body>
        <div id="idButtons" class="paragraph"></div>
        <h1>High Plant Production Report</h1>

        <table width="60%">
            <tr>
                <th>Item</th><th>Description</th><th>Quantity</th>
            </tr>
            <tr>
                <td>1</td>
                <td>Part A</td>
                <td data-igrx="{attr:'get',tag:'part_a'}">0.00</td>
            </tr>
            <tr>
                <td>2</td>
                <td>Part B</td>
                <td data-igrx="{attr:'get',tag:'part_b'}">0.00</td>
            </tr>
            <tr>
                <td>3</td>
                <td>Part C</td>
                <td data-igrx="{attr:'get',tag:'part_c'}">0.00</td>
            </tr>
            <tr>
                <td>4</td>
                <td>Part D</td>
                <td data-igrx="{attr:'get',tag:'part_d'}">0.00</td>
            </tr>
        </table>
        <h1>Production Log</h1>
        <table width="60%">
            <tr>
                <th width="28%">Description</th>
                <th width="18%">Density</th>
                <th width="18%">Pressure</th>
                <th width="18%">Level</th>
                <th width="18%">Weight</th>
            </tr>
            <tr>
                <td>Time/Unit</td>
                <td>(kg/m3)</td>
                <td>(bar g)</td>
                <td>(m)</td>
                <td>(kg)</td>
            </tr>
            <tr>
                <td data-igrx="{attr:'log',tag:'app.currentTime'}"></td>
                <td data-igrx="{attr:'log',tag:'part_a'}">0.00</td>
                <td data-igrx="{attr:'log',tag:'part_b'}">0.00</td>
                <td data-igrx="{attr:'log',tag:'part_c'}">0.00</td>
                <td data-igrx="{attr:'log',tag:'part_d'}">0.00</td>
            </tr>
            <tr>
                <td>Summary</td>
                <td>Avg</td>
                <td>Max</td>
                <td>Min</td>
                <td>Sum</td>
            </tr>
            <tr>
                <td></td>
```

```
                <td>0.00</td>
                <td>0.00</td>
                <td>0.00</td>
                <td>0.00</td>
            </tr>
        </table>

    <script type="text/javascript" src="./system/scripts/igrX.js"></script>
    <script type="text/javascript">
        var rpt = igrx.report();
    </script>
    </body>
</html>
```

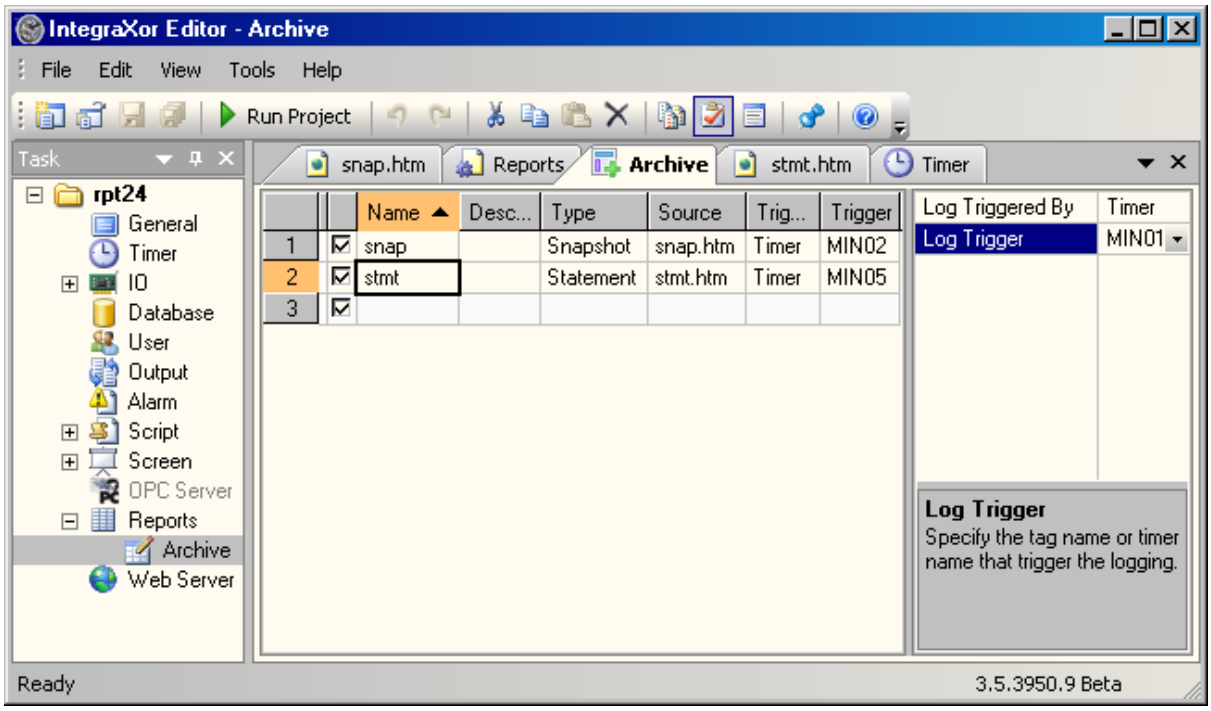
It's common to have another additional summary at the last row of statement report. And the built-in calculations that available for use are like average (avg), maximum (max), minimum (min) & summation (sum). Set the attribute (attr) to the desired calculation syntax in the bracket above. Below is the sample after adding in the additional summary row that configured with summary syntax.

```
<!DOCTYPE HTML>
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <title>IntegraXor Report</title>
        <link rel="stylesheet" type="text/css"
                href="system/styles/igrX.css" />
    </head>
    <body>
        <div id="idButtons" class="paragraph"></div>
        <h1>High Plant Production Report</h1>

        <table width="60%">
            <tr>
                <th>Item</th><th>Description</th><th>Quantity</th>
            </tr>
            <tr>
                <td>1</td>
                <td>Part A</td>
                <td data-igrx="{attr:'get',tag:'part_a'}">0.00</td>
            </tr>
            <tr>
                <td>2</td>
                <td>Part B</td>
                <td data-igrx="{attr:'get',tag:'part_b'}">0.00</td>
            </tr>
            <tr>
                <td>3</td>
                <td>Part C</td>
                <td data-igrx="{attr:'get',tag:'part_c'}">0.00</td>
            </tr>
            <tr>
                <td>4</td>
                <td>Part D</td>
                <td data-igrx="{attr:'get',tag:'part_d'}">0.00</td>
            </tr>
        </table>
        <h1>Production Log</h1>
        <table width="60%">
            <tr>
                <th width="28%">Description</th>
                <th width="18%">Density</th>
                <th width="18%">Pressure</th>
                <th width="18%">Level</th>
                <th width="18%">Weight</th>
            </tr>
            <tr>
                <td>Time/Unit</td>
```

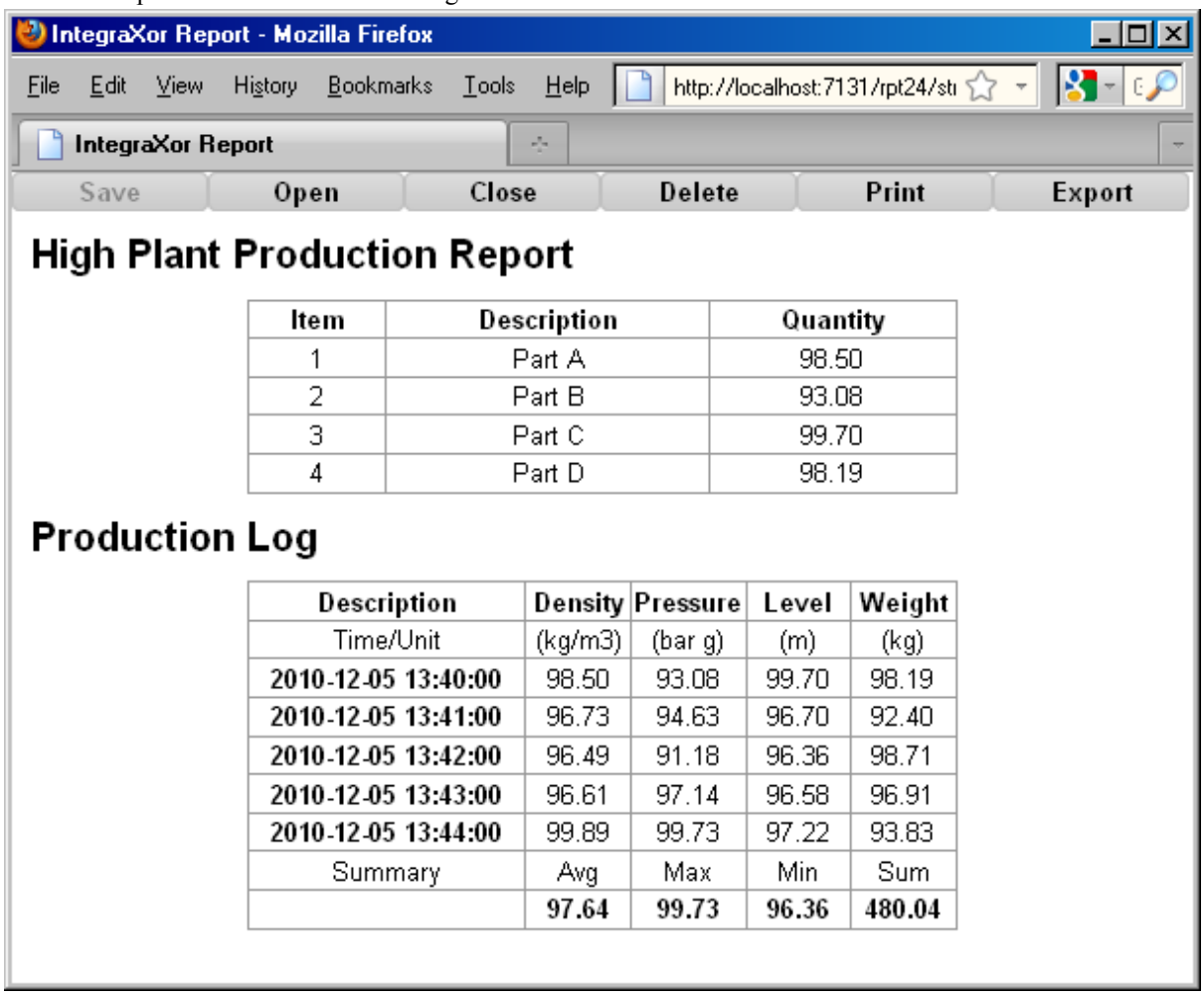
```
                <td>(kg/m3)</td>
                <td>(bar g)</td>
                <td>(m)</td>
                <td>(kg)</td>
            </tr>
            <tr>
                <td data-igrx="{attr:'log',tag:'app.currentTime'}"></td>
                <td data-igrx="{attr:'log',tag:'part_a'}">0.00</td>
                <td data-igrx="{attr:'log',tag:'part_b'}">0.00</td>
                <td data-igrx="{attr:'log',tag:'part_c'}">0.00</td>
                <td data-igrx="{attr:'log',tag:'part_d'}">0.00</td>
            </tr>
            <tr>
                <td>Summary</td>
                <td>Avg</td>
                <td>Max</td>
                <td>Min</td>
                <td>Sum</td>
            </tr>
            <tr>
                <td></td>
                <td data-igrx="{attr:'avg',tag:'part_a'}">0.00</td>
                <td data-igrx="{attr:'max',tag:'part_b'}">0.00</td>
                <td data-igrx="{attr:'min',tag:'part_c'}">0.00</td>
                <td data-igrx="{attr:'sum',tag:'part_d'}">0.00</td>
            </tr>
        </table>

    <script type="text/javascript" src="./system/scripts/igrX.js"></script>
    <script type="text/javascript">
        var rpt = igrx.report();
    </script>
    </body>
</html>
```

7.2.3. Archiving Report

Similar to Snapshot report, statement report requires a timer to be specified as generation time. On top of it, additional timer is needed for interval logging.



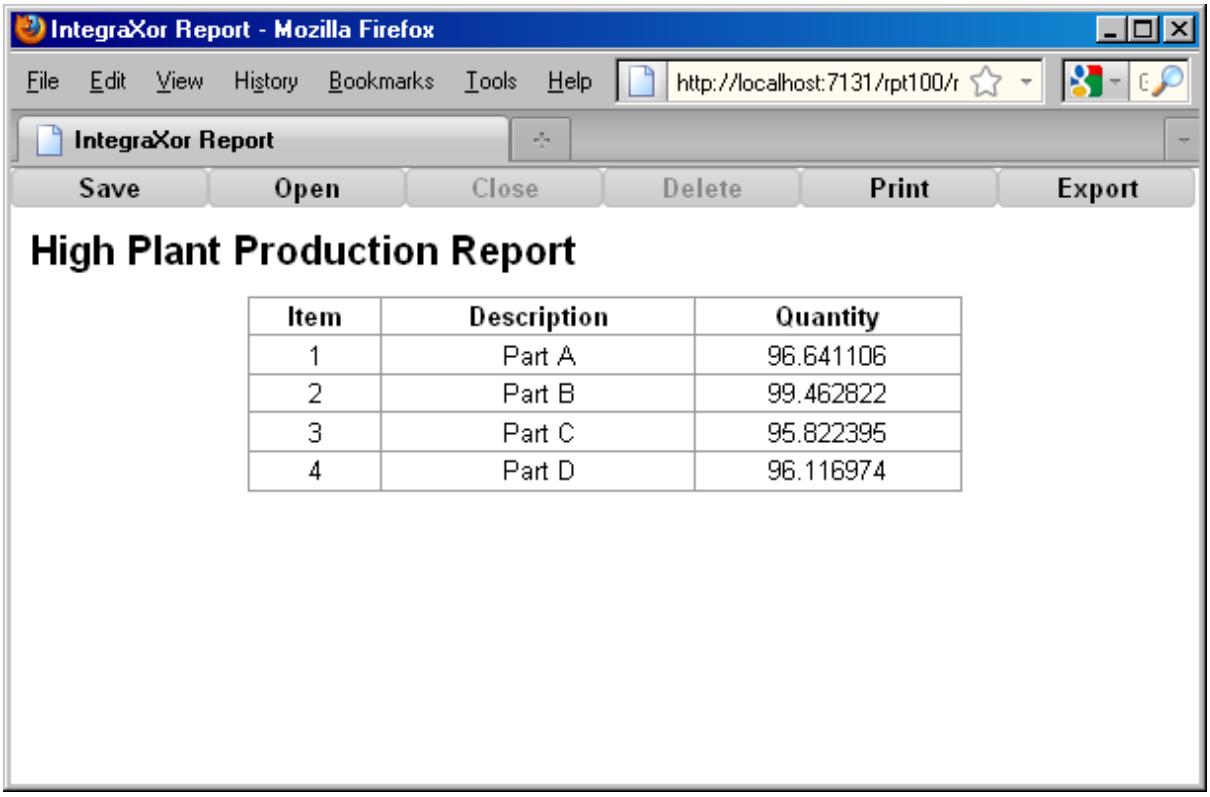
Statement report needs to have interval log timer.


IntegraXor Report - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:7131/rpt24/sti
IntegraXor Report
Save Open Close Delete Print Export
High Plant Production Report
Item Description Quantity
1 Part A 98.50
2 Part B 93.08
3 Part C 99.70
4 Part D 98.19
Production Log
Description Density Pressure Level Weight
Time/Unit (kg/m3) (bar g) (m) (kg)
2010-12-05 13:40:00 98.50 93.08 99.70 98.19
2010-12-05 13:41:00 96.73 94.63 96.70 92.40
2010-12-05 13:42:00 96.49 91.18 96.36 98.71
2010-12-05 13:43:00 96.61 97.14 96.58 96.91
2010-12-05 13:44:00 99.89 99.73 97.22 93.83
Summary Avg Max Min Sum
97.64 99.73 96.36 480.04

A completed running report with data feed.

7.3. Report Toolbar Menu

Report toolbar menu will be automatically insert into `<div id="idButtons" class="paragraph"></div>`, normally placed on top of the page when the layout is viewed on screen. The menu consists of several file management buttons, Save, Open, Close, Print, Export and Delete.


IntegraXor Report - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:7131/rpt100/r
IntegraXor Report
Save Open Close Delete Print Export
High Plant Production Report
Item Description Quantity
1 Part A 96.641106
2 Part B 99.462822
3 Part C 95.822395
4 Part D 96.116974

Report Toolbar will be Automatically Added.

Report Toolbar that available for use in realtime, Save,& Open.

Report Toolbar that available for use in archive, all available except Save.

Save — Save button is disabled in archive mode. When Save button is clicked in real-time mode, all live data will be archived into database instantly with timestamp, then automatically loaded and hence enter archive mode that showing last saved data. And Close button will then enabled so end user could go back to real-time mode by closing last saved archive.

Open — Open button is enable at all time. When Open button is clicked either in real-time or archive mode. A dialog box shall pop up and display the default directory, prompting operator or end user to select an archive for viewing. Once the archive is open, background shall turn gray to indicate that report viewer has entered into archive mode and data on screen will stop updating. Close button will be enabled in archive mode for operator to put the system back to auto refresh real-time mode.

Close — Clicking Close shall put the system back to real-time mode.

Print When Print button is clicked, an ordinary printing dialog box shall pop up, so user could print to the desired printer. Print button is available in archive mode.

Export When Export button is clicked, A dialog box like Save As shall pop up with additional option, so end user can pick to export into HTML file, Comma Separated Values or XML file. Export button is available in archive mode.

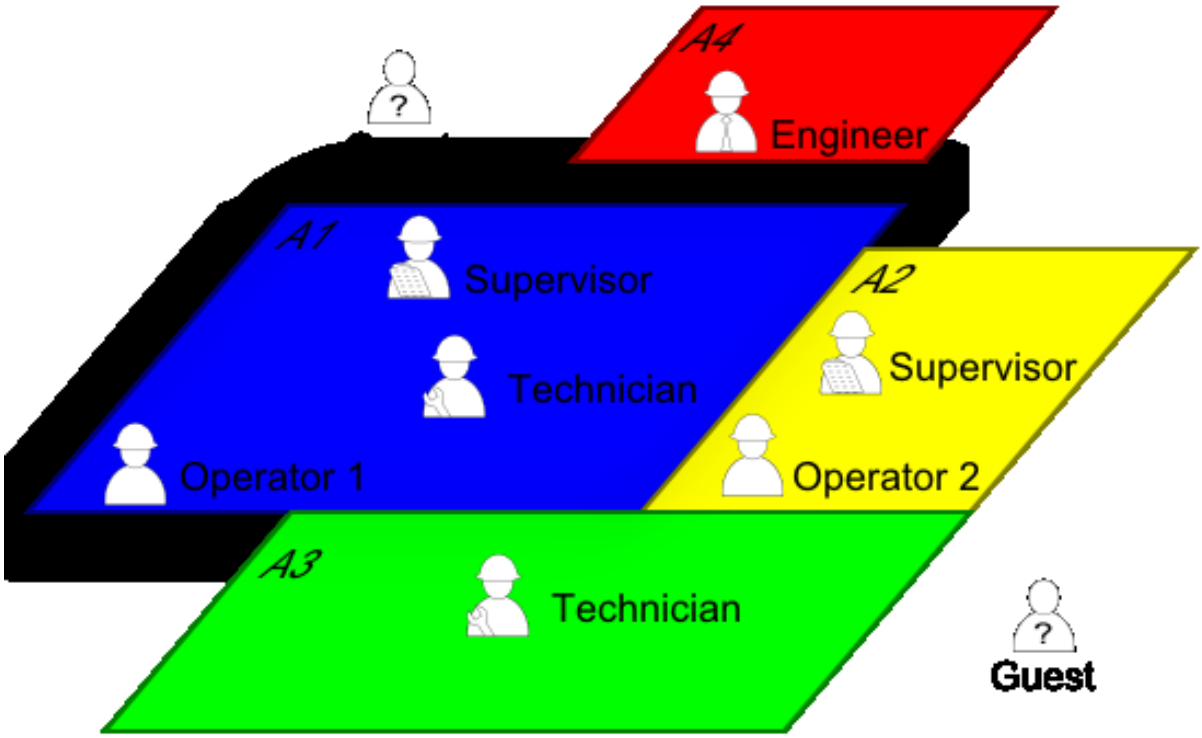
Delete Delete button is available in archive mode, when an archive is being viewed. When Delete button is clicked, the currently viewed archive will be prompted to be deleted. System shall automatically go to real-time mode after deleting the archive.

Chapter 8. Security

Security in IntegraXor SCADA system is extended into three perspectives, i.e. data, screen & action. "Data" means runtime tags or archive, "Screen" is mimic page, and actions refers to Acknowledging or configuring alarm, Printing, Exporting and Saving. The protection on Screen is limited to Read Control, & the protection for Action is limited to Write control, and the Data integrity can be set in both Read & Write control.

The login authentication can be performed based on the location where user is logging in to ensure tighter security, and then the restriction of logging sessions is versatile enough that can be further refined into role and/or user basis. i.e. maximum concurrent logging of session for one particular role, maximum multiple logging session for single user, and combination of both types of logging session.

As compare to simple read permission, write permission (which could well mean making changes or taking control) has even greater configuration details to allow smoother operation of a SCADA system. For instance, an exclusive write permission can be set onto one particular role, or a concurrent write permission can be achieved by granting write permission to several roles, whereby write operation will be performed based on first come first serve basis. Moreover, to allow the greatest possible security configuration, the write permission can be swap with another user or even pass down to another user when logging out.

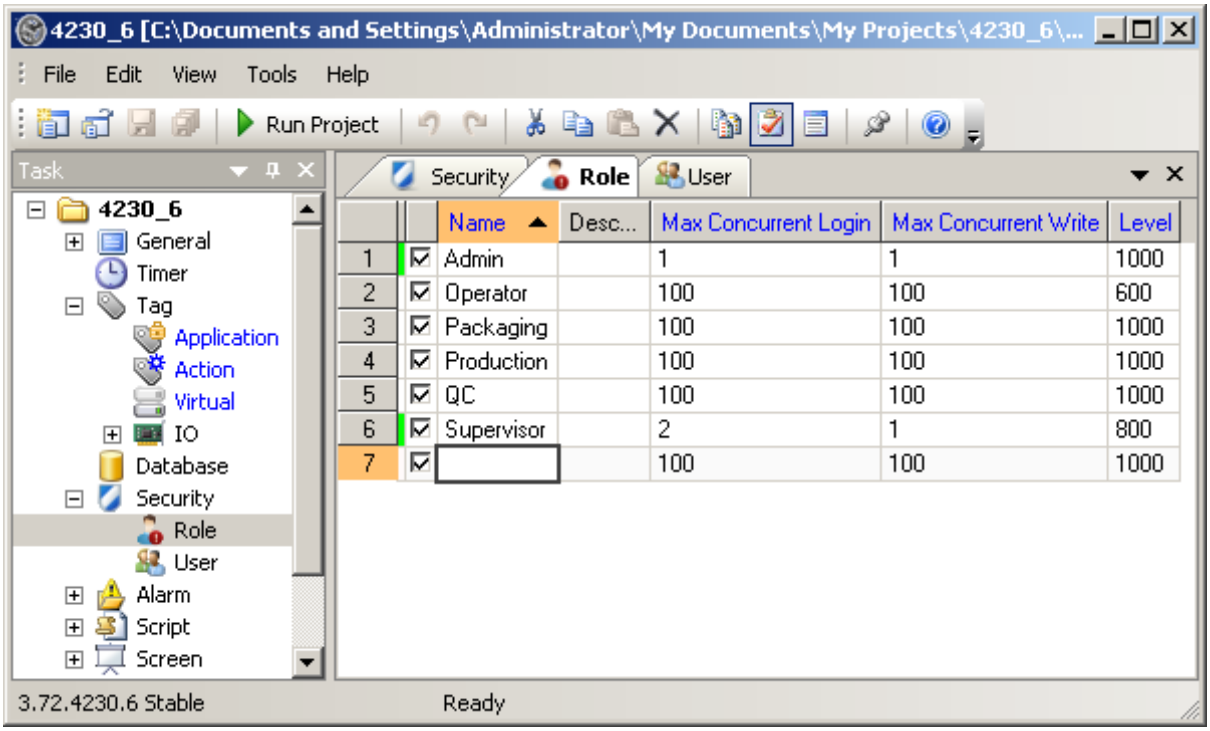


Security Access based on Role/Area Control

8.1. Security Role

Security architecture in IntegraXor SCADA is using Role-Based Access Control (RBAC), user admin shall start the design by defining the security role as a whole. For instance, a simple security hierarchy in a factory can be split into Engineer, Supervisor, Technician & Operator. These defined roles can then be associated to individual tags or database for data integrity, or Screen for page visibility, or Action for the control operation.

Role for user can be added as many as the system required, a user can be assigned to one or more roles, and a same role can also be applied to multiple users. Be aware when adding role onto tag, as that will restrict access for guest. Putting role on users is like granting them key, putting role on tag/database means adding lock onto those associated equipments. They must be matched in order to pass the security authentication.



Security Access based on Role/Area Control

Each security role could be granted with maximum concurrent login and maximum concurrent write. The Level can be used if hierarchy control is involved. The configured value for maximum concurrent login will determined the maximum allowable logins for a particular role. If the role carried by a user has reached the limit, login will be prohibited. However, for user who assigned with multiple roles, login is permitted when at least one of the role did not reach the login limit. Whereas maximum concurrent write will limit the distribution of write permissions to a particular role. When the limit is reached, user will not attain write permissions for the particular role. Normally lesser concurrent shall be set to avoid conflict in control operation.

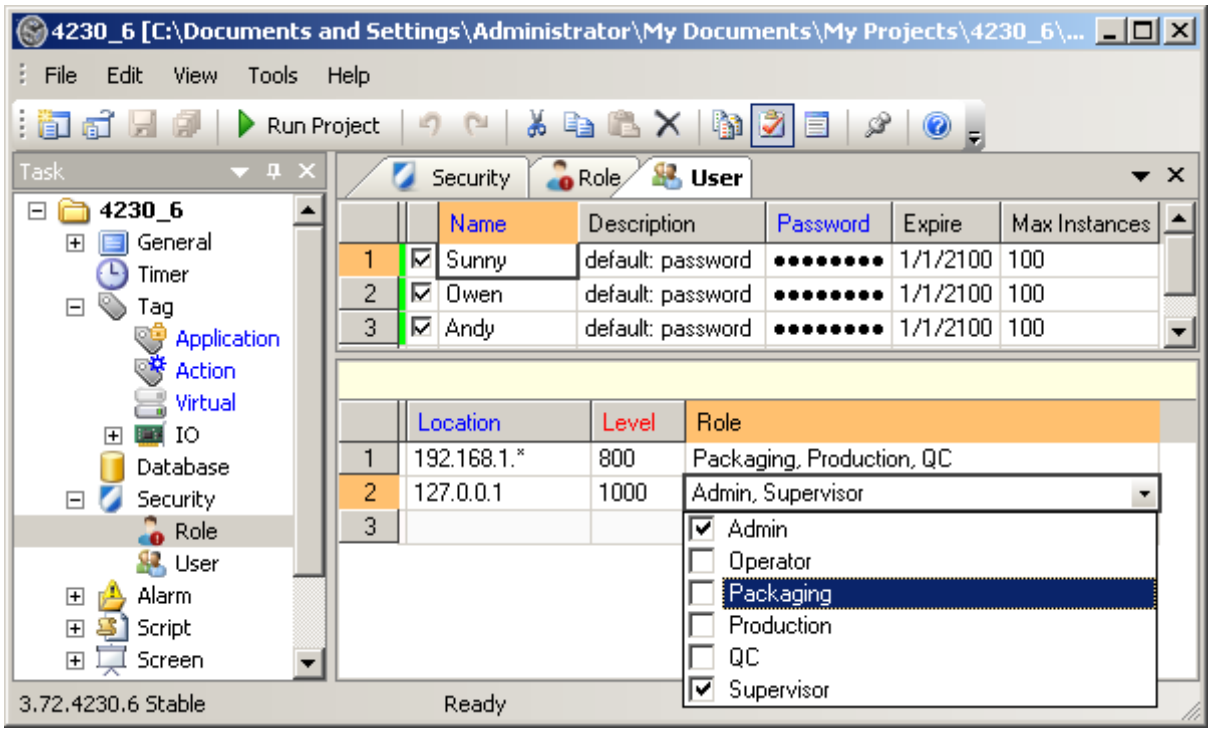
When a user logged into the system, the user will carry all the access rules defined by the role. Since the access permissions is not directly assigned to the user, management of individual user rights becomes a matter of simply assigning appropriate roles to the user's account. Thus this simplifies common operations, such as adding a user, or changing a user's department/location.

Note

Formally Role is being called "Label". It was renamed as "Role" to describe the context better.

8.2. User Configuration

Administrator could create as many user as per requirement with different sets of role. User's role can be seen as a user group in security context. In User configuration, a SCADA operation user can be created and setting the initial password, expiry date and maximum login instances or session. If this instance is set to value of one (1), then login from multiple locations is prohibited.



User Security Configuration

Once a user is created, security role can then be granted to as per requirement. Note that the multiple role can be assigned to a single user by using comma as separator. Furthermore, the Location IP can be associated to each role for specific user for tighter security control. Single `*` is referring to any location login, `192.168.1.*` means allowing login from any IP start with `192.168.1`. Multiple IP addresses can be entered by splitting with comma.

The numbers of role can be associted without any restriction, but it will only be granted to the login location specified unless it's entered as *. This means any data or content protection that associated with the role can only be accessible when user login from the associated location. For instance, the admin or system security planner may want to allow an Operator to start a pump from site control room but can only check the pump status from office terminal, Supervisor is allowed to start a pump from both site & office but only can check the pump status from home, and then Engineer may be allowed to start a pump from all 3 locations.

Specifying IP Address into Location shall greatly enhance the SCADA security by blocking any intrusion from unknown locations, especially for SCADA system that open to Internet access. For instance, a standalone SCADA system shall be entered with 127.0.0.1 so no remote client is allowed, and for local area network access only should be entered with same range of IP like 192.168.1.*. And if the system is meant for Internet access, administrator shall check out the country IP range to restrict the access from own country and any other possibly needed countries only.

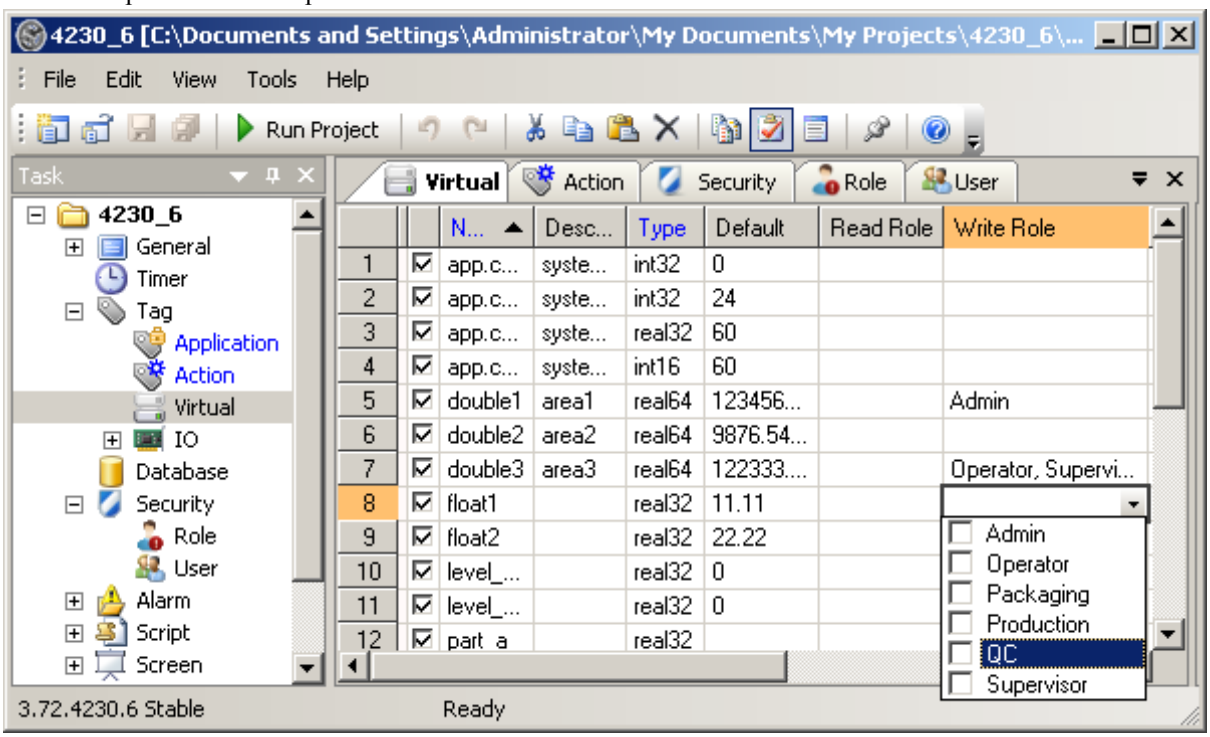
Warning

The Level in User entry shall avoid to be used as it is depricated and is kept for legacy support only.

8.3. Protection

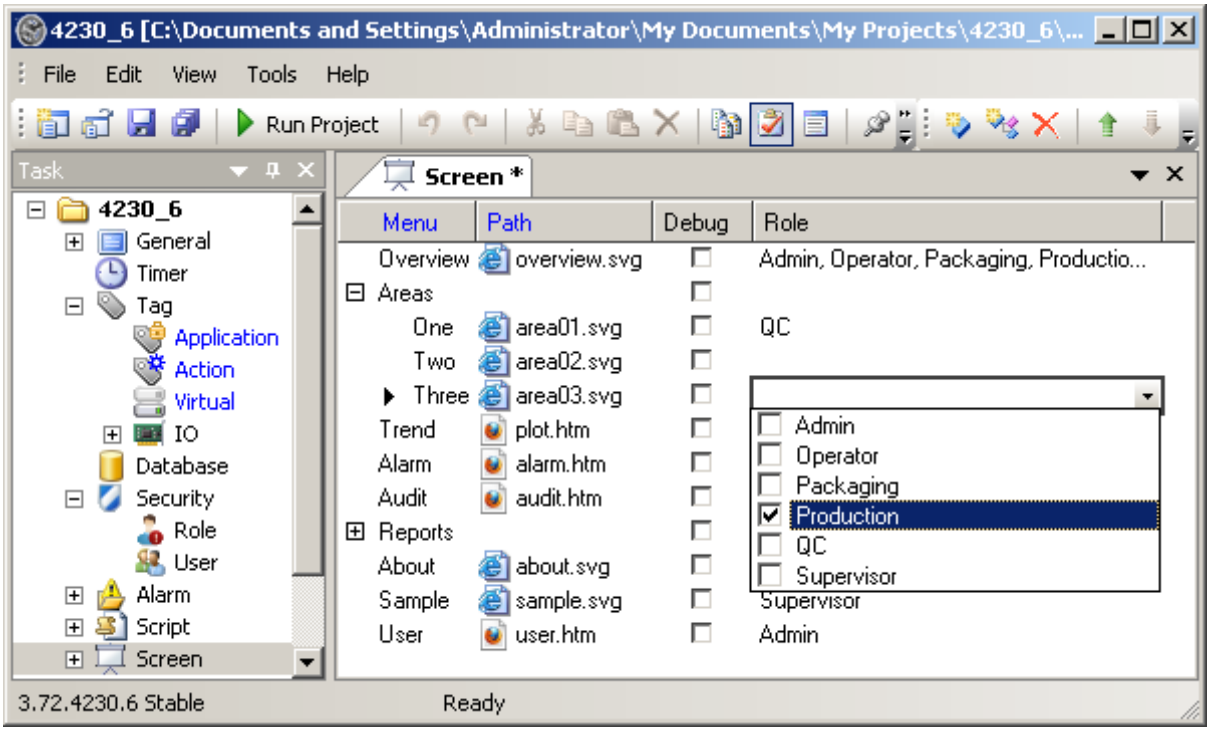
Security protection can be applied onto tags, databases, screens access and also operation action. To specify security role for each tag is straightforward, simply enter desired role(s) for each tag. The security access for both read and write is default to no security (world access). And they can be set to either similar or different role set for read and write on single tag. Similarly, the same way of adding security can be extended to Action tags

so certain operations can be protected.



Each & individual tags can be protected with different sets of rules for both read and write.

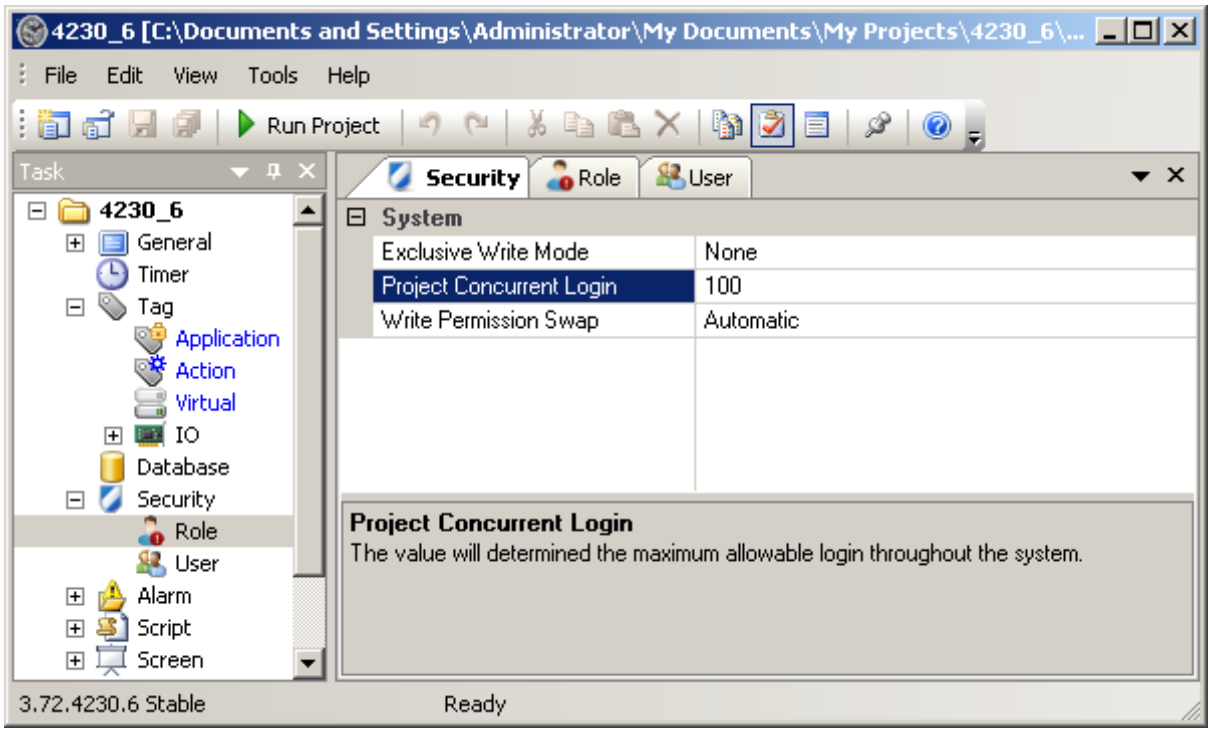
Protection for screen access is bound to opening only, in another word, a mimic screen can be restricted by being viewed from a user based on the role assigned. However, the data within the screen is still fallback and being protected based on the individual tags security control rather than based on the mimic screen itself.



Mimic screen can be restricted to the specific group of user.

8.4. Additional Security Configuration

Greater flexibility in controlling the security can be achieved by configuring additional security setting as a whole. To ensure no excessive login in one system with limited traffic bandwidth, enter the maximum number into Project Concurrent Login.



Additional details for security control.

The option of Exclusive Write Mode is meant to allow only one role to have write permission throughout the system. If the option 'Highest role's level' is selected, even if a tag is associated to multiple roles, only the role with highest role's level is able to modify the tag value. For example, given the condition that 'tagA' is associated with role Admin which set with level 1000 and Supervisor which set with level 500. When Susan with 'Supervisor' role logged into the system, she will have the write permission for 'tagA'. However, once Eric with 'Admin' role logged into the system, write permission will be given to Eric, due to higher role level. Susan will have to wait for Eric to logout before she is permitted to write on 'tagA' again.

The option of Write Permission Swap is to define the method of assigning write permission. When 'Automatic' mode is selected, assigning write permission will be executed immediately and automatically without the acknowledgment of receiving party. When 'Prompt' mode is selected, consent of the receiving party is required to complete the process. If the receiving party failed to respond within a specific duration, the request will then be disregarded.

To allow automatically pass over the Write Permission, Write Permission Pass Over When Logout will allow Administrator to choose how the permission shall hand over automatically during logout, it can pass over based on IP priority, or based on first login or last login.

Chapter 9. Communication

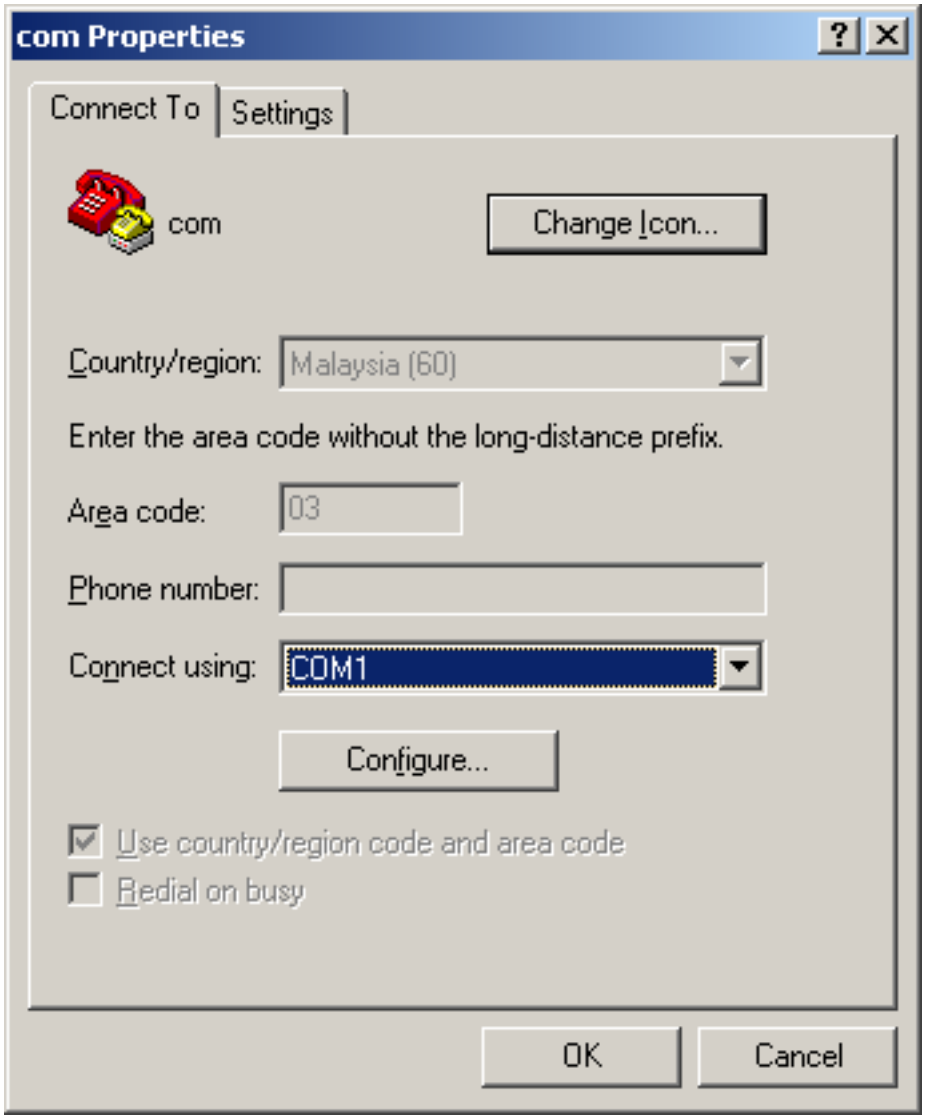
Establishing communication could be problematic especially on serial communication. It's advised that all aspects are tested before starting tedious data entry work. For instance, test through all needed communication protocols, all required data-types shall be confirmed communicating well, read and write operation is achievable. This section which may help to narrow down the troubleshooting scope when several direct attempts failed.

Tip

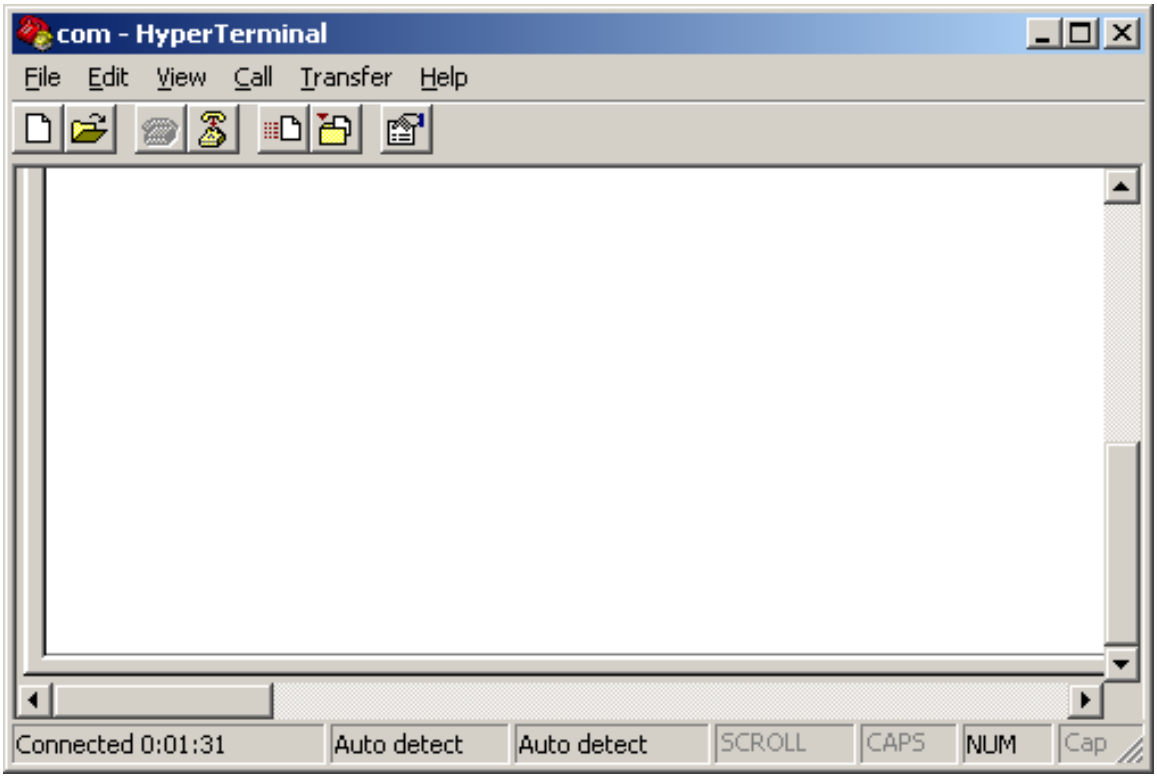
The evaluation IntegraXor software is fully functional until the demo period is expired so unregistered license has no impact in establishing communication.

9.1. Setting Communication Port

1. First and foremost is to check for serial port availability, engineer who's using Ethernet connection could skip to next step. Nowadays most laptops and PCs do not equipped with COM port and USB to Serial converter is needed to emulate COM port. COM port only allows single application to be occupied at any one time. To test if the COM port driver is successfully installed, open up Hyper Terminal from Windows Start menu \ Acccesorries \ Communications. Check that the desired COM port is available as one of the option for connection.



COM 1 is available for connection



Bottom left corner showing 'Connected' status. Note that this only means the port has been successfully open but doesn't mean Hyper Terminal has started communicating with any external devices.

Important

Ensure Hyper Terminal is disconnected or closed after confirming the availability of port. Otherwise it may occuppied the port and obstruct other programs.

2. Ensure physical connectivity is the most essential step. Try to use device supplied programming software or third party software to establish the communication between targeted PC and device. For instance, third party OPC server could be used for polling to confirm that the physical link is properly connected.

   Serial link requires communication parameters to be *exactly* matched between SCADA and devices. Engineer must confirm the parameter settings in the devices and configure exactly the same for server, i.e. Baud Rate, Data Size, Parity & Stop Bit. e.g. 9600 8 none 1.

   TCP/IP link would be very much easier to setup as compare to serial link. Anyhow both of them still need to configure the node address and eventually a list of data address. Engineer must know the configured node address for the device and the available list of data address and its data type, and again they must be *exactly* matched. Before the communication is established, only one integer data address should be entered in order to avoid confusion.

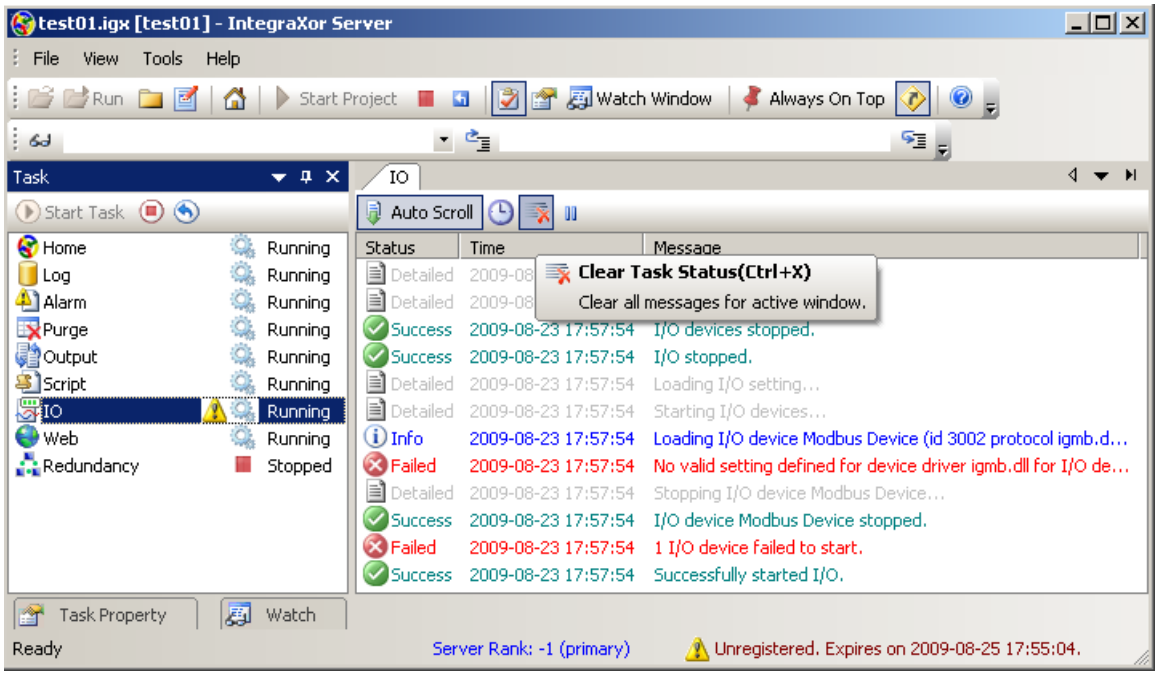
Important

Ensure all tools and programs that opened for testing are closed before proceeding to the next step.

3. Always try to narrow down the scope of the troubleshooting. Avoid having any converter in between, unless it's inevitable.

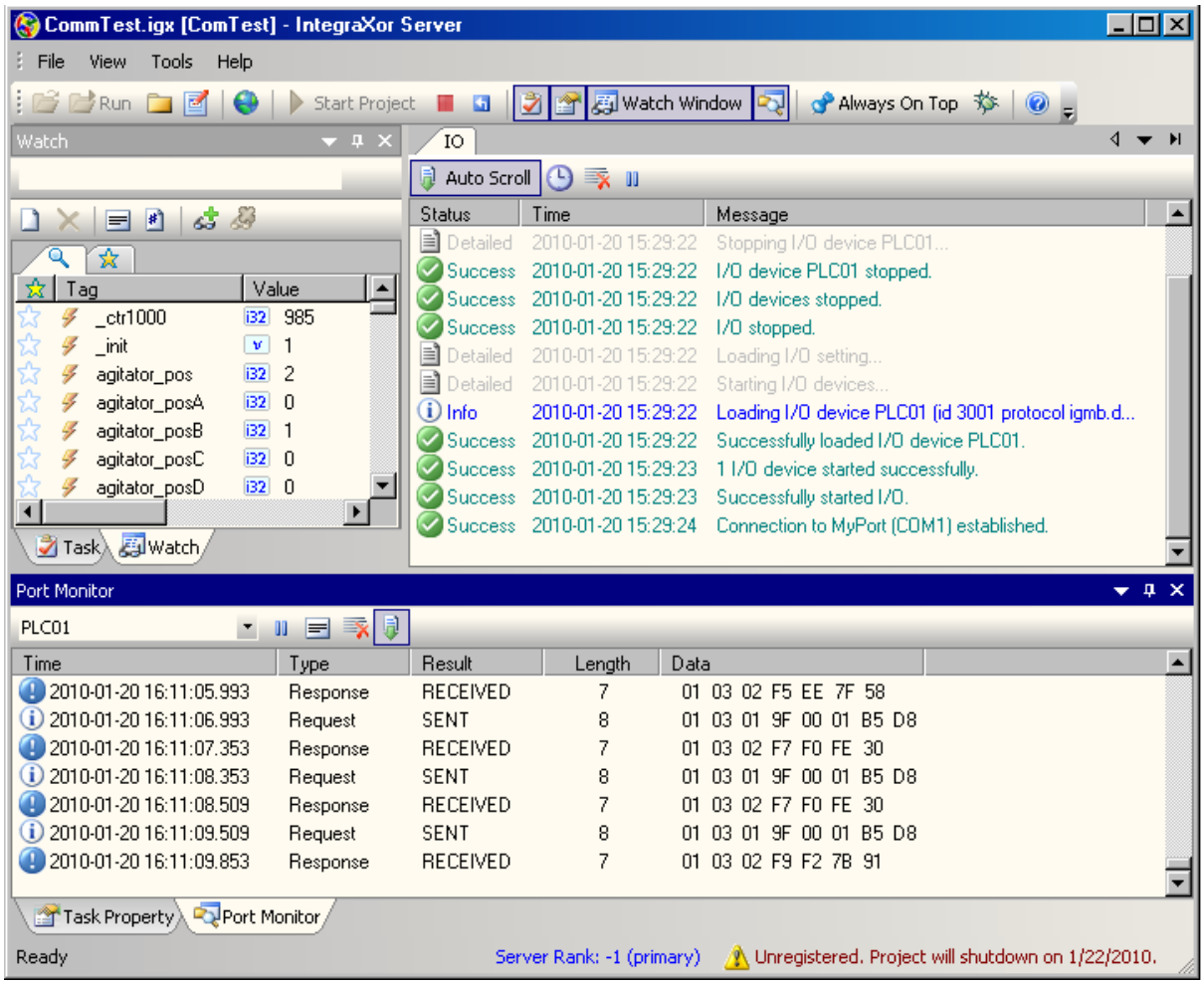
4. If the programming tool or third party software managed to establish the communication, it's pretty clear

that the problem is within the configuration. Firstly, open up the server GUI, click on the IO Task, and check out its status log. If there's any problem in the communication, an exclamation mark shall appear next to the IO icon in the Task Window. And red text describing the error shall be logged. If too many messages are logged, click on Clear Task Status to remove old messages, then click on Restart Task to start up IO again. And latest IO status shall appear in the status log.


test01.igx [test01] - IntegraXor Server
File View Tools Help
Run Start Project Watch Window Always On Top
Task
Start Task
Home Running
Log Running
Alarm Running
Purge Running
Output Running
Script Running
IO Running
Web Running
Redundancy Stopped
IO
Auto Scroll
Status Time Message
Clear Task Status(Ctrl+X)
Clear all messages for active window.
Success 2009-08-23 17:57:54 I/O devices stopped.
Success 2009-08-23 17:57:54 I/O stopped.
Detailed 2009-08-23 17:57:54 Loading I/O setting...
Detailed 2009-08-23 17:57:54 Starting I/O devices...
Info 2009-08-23 17:57:54 Loading I/O device Modbus Device (id 3002 protocol igmb.d...
Failed 2009-08-23 17:57:54 No valid setting defined for device driver igmb.dll for I/O de...
Detailed 2009-08-23 17:57:54 Stopping I/O device Modbus Device...
Success 2009-08-23 17:57:54 I/O device Modbus Device stopped.
Failed 2009-08-23 17:57:54 1 I/O device failed to start.
Success 2009-08-23 17:57:54 Successfully started I/O.
Task Property
Watch
Ready
Server Rank: -1 (primary)
Unregistered. Expires on 2009-08-25 17:55:04.
IO task must be checked during communication troubleshooting

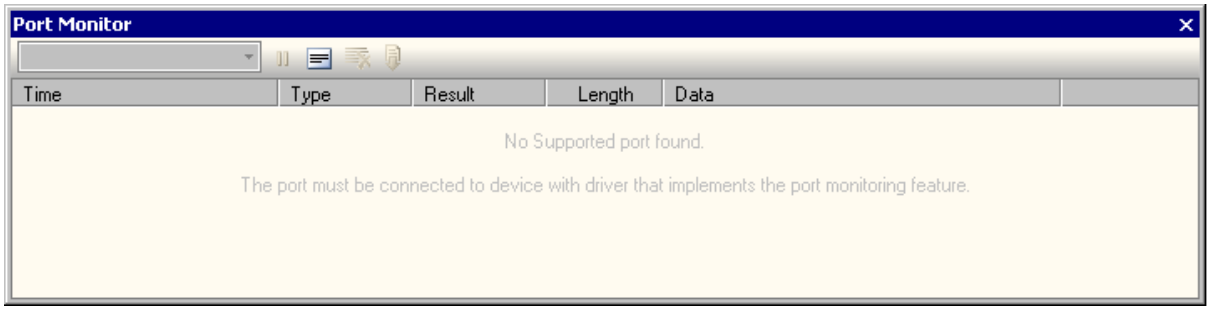
9.2. Port Monitor

IntegraXor comes with built-in Port Monitor utility. It aid in advanced communication troubleshooting whereby engineer could monitor the communication traffic and analyse its bits and bytes of data during runtime.


CommTest.igx [ComTest] - IntegraXor Server
File View Tools Help
Run Start Project Watch Window Always On Top
Watch
Tag Value
_ctr1000 985
_init 1
agitator_pos 2
agitator_posA 0
agitator_posB 1
agitator_posC 0
agitator_posD 0
Task Watch
IO
Auto Scroll
Status Time Message
Detailed 2010-01-20 15:29:22 Stopping I/O device PLC01...
Success 2010-01-20 15:29:22 I/O device PLC01 stopped.
Success 2010-01-20 15:29:22 I/O devices stopped.
Success 2010-01-20 15:29:22 I/O stopped.
Detailed 2010-01-20 15:29:22 Loading I/O setting...
Detailed 2010-01-20 15:29:22 Starting I/O devices...
Info 2010-01-20 15:29:22 Loading I/O device PLC01 (id 3001 protocol igmb.d...
Success 2010-01-20 15:29:22 Successfully loaded I/O device PLC01.
Success 2010-01-20 15:29:23 1 I/O device started successfully.
Success 2010-01-20 15:29:23 Successfully started I/O.
Success 2010-01-20 15:29:24 Connection to MyPort (COM1) established.
Port Monitor
PLC01
Time Type Result Length Data
2010-01-20 16:11:05.993 Response RECEIVED 7 01 03 02 F5 EE 7F 58
2010-01-20 16:11:06.993 Request SENT 8 01 03 01 9F 00 01 B5 D8
2010-01-20 16:11:07.353 Response RECEIVED 7 01 03 02 F7 F0 FE 30
2010-01-20 16:11:08.353 Request SENT 8 01 03 01 9F 00 01 B5 D8
2010-01-20 16:11:08.509 Response RECEIVED 7 01 03 02 F7 F0 FE 30
2010-01-20 16:11:09.509 Request SENT 8 01 03 01 9F 00 01 B5 D8
2010-01-20 16:11:09.853 Response RECEIVED 7 01 03 02 F9 F2 7B 91
Task Property Port Monitor
Ready
Server Rank: -1 (primary)
Unregistered. Project will shutdown on 1/22/2010.

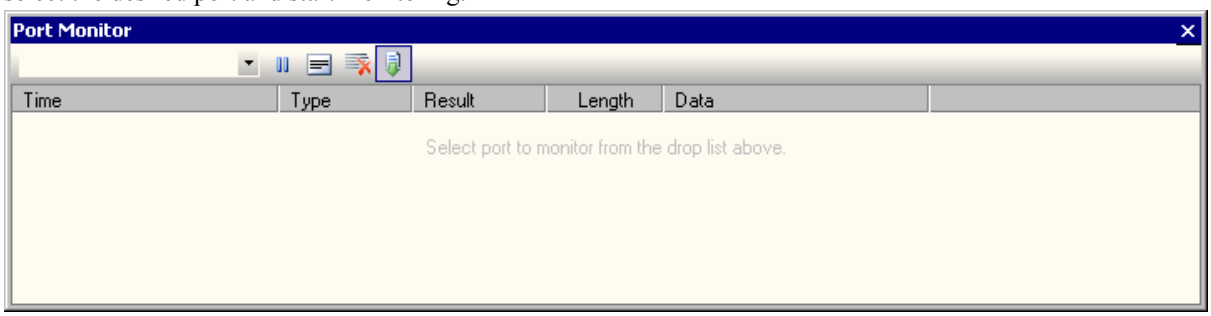
Port Monitor showing data traffic for advanced analysis

The Port Monitor window/pane can be viewed leaving it embeded in the server main window, or even dragged out and float independently.


Port Monitor
Time Type Result Length Data
No Supported port found.
The port must be connected to device with driver that implements the port monitoring feature.

When no port is available for monitoring or when project is not running.

Using Port Monitor is straightforward, simply configure a communication port as usual, start the project and select the desired port and start monitoring.



Select a port to monitor

9.3. Modbus Troubleshooting for SCADA

On top of basic communication port setting, it's necessary to understand basic Modbus protocol before trying to establish a Modbus link. Unlike other documentation for Modbus, this document focus on the knowledge needed for establishing the communication rather than implementing it. This perspective will be more relevant for System Integrator Engineer.

Modbus is a very old protocol that has too many interpretations and implementations by various vendors, making it one of the widely used but complicated protocol to establish at the beginning. Too many aspects need to be of concern hence plug and play is almost impossible for two applications from different vendors for the first time. User should start by entering 1 or 2 tags with basic data type like Holding Register (int16) or Output Coil (bool), and slowly adding more tags before advancing to more complicated data type like real32 or long32.

Several terminologies have been used in industry to refer to the same thing which created more confusion. This document used the most precise and non-ambiguous term, but other vendor terms will also be mentioned.

RTU/ASCII Mode

Modbus can be used in RTU or ASCII communication mode. RTU is mostly used as compare to ASCII since it's transmitted in binary and hence faster. On the other hand ASCII mode is rarely being used and will only be chosen when the communication speed (baud rate) can't go too fast due to distance or communication device (e.g. modem) limitation. They cannot be used concurrently in one network.

Important

The data byte size for Com Port setting is 8 for RTU and 7 for ASCII.

Master/Slave(s)

The relationship of Master and Slave in Modbus context is like Client and Server in computer network. Modbus network in serial communication can only have one Master with one, or many other Slaves where it's called "Multidrop" network. Normally the Master is a PC or PLC and the slave(s) could be some measuring devices or sensors. The Master will initiate all data polling or writing sequence and the Slave being called will then respond accordingly. This is totally opposite of computer network whereby normally one server will serves many other clients, despite it's similar that the client will initiate all the transactions.

Note

Single master restriction only applicable to Serial Communication like RS232/422/485. Modbus over TCP/IP network could support multiple Masters and of course multiple Slaves. Multidrop network topology only available for RS485 & RS422 network configuration, physical layer of RS232 do not support multi-drop network.

Node ID

Modbus device can be connected in multi-drop (series) network. And every Modbus device must be assigned with an ID for identification, even if there's only one device is being connected. Normally it's default to 1.

Note

Node ID also being called Device ID, Slave ID, Device Address, Slave Address or even Modbus

Address. Some devices may even contain several Node Addresses, which being called virtual device or virtual node address.

Data Address

Every data must has its own address. Data Address also being called as Register or simply as Address. Data Address indexing may be different from one vendor to another, in another word, address 1000 for one vendor might be 1001 or 999 for another vendor. Offset value of +1 or -1 need to be entered.

In PLC implmentation, prefixes are normally added for datatype differentiation. 4x for Output Register, 3x for Input Register, 0x for Output Coil and 1x for Input Coil. However this introduces restriction in modern SCADA system and will normally disregard. For instance a device with 6 digits address, 400100 in device need to be entered as 100 with required datatype (int, float etc) in SCADA. Likewise a device with 5 digits address, 10012 in another device will need to enter as 12 only in SCADA with required datatype (boolean+input).

Address Arrangement

The address arrangement is flexible in Modbus devices. The starting address for any data type is not fixed, for instance it can be 1, 1000, 4000 or any other value. There's no fixed patent on how's the addresses will be arranged. And the addresses are not necessary need to be put into one range, they can be scattered around like 1000~2000 and 4000~10000. The address arrangement is completely depends on the device manufacturer and must refer to its documentation.

Data Type

They are only four fundamental data types in Modbus, i.e. Output Coil (0x), Input Coil (1x), Holding Register (4x) & Input Register (3x). Both Output Coil (aka status) & Input Coil (aka status) are also boolean but further split into input and output type. Similarly both Input Register & Output Register are also 16 bit Integer but further split into input and output category. Input Coil and Input Register are strictly tied to device's physical wire connection, on the other hand, Output Coil and Output Register can be tied or not tied to device's physical wire connection. When an address is not tied to physical link, it's treated as memory variable. So most of the time the addresses being used are output type.

In order to get more than four fundamental types of data, like floating point (float32) and long analogue (int32), two Output Registers need to be used to achieve this purpose. When a list of floating point data or long analogue data being listed in sequence, some devices like Modicon PLC will have the addresses skipped in even number or odd number, but some may just have it in sequence without any skipping of address. User must figure this out based on the device's manual or even trial & error.

Some devices can further support double floating point (float64) or double long integer (int64), the address skipping issue is again depends on the device's manufacturer. Moreover, some manufacturers even have proprietary data type like Time Element that contains up to 8 integers for one single address.

RW or RO

One data location can be implemented as Read/Write enabled, or just Read Only. This is totally depends on the individual vendor and must refer its

documentation. Normally Input Register (3x) and Input Coil (1x) shall be implemented as Read Only.

Word Swapping

For user that uses either floating point or long data type, they need to understand swapping of words (int16) (or even bytes (int8) for some devices) in Modbus communication since both float and long data type constitute of two or more registers (int16). Different vendor has different implementation on handling binary data sorting, which is either Little Endian or Big Endian. This mean float or long data type can be sorted in the form of A+B or B+A whereby A and B are int16. And for double float or long, the combination could be extended to A+B+C+D, B+A+D+C, C+D+A+B or D+C+B+A.

Since the term used for the definition of the condition is different from all vendors, user need not to have deep understanding of scenario but simply need to try out swapped or direct setting and observe the outcome.

Note

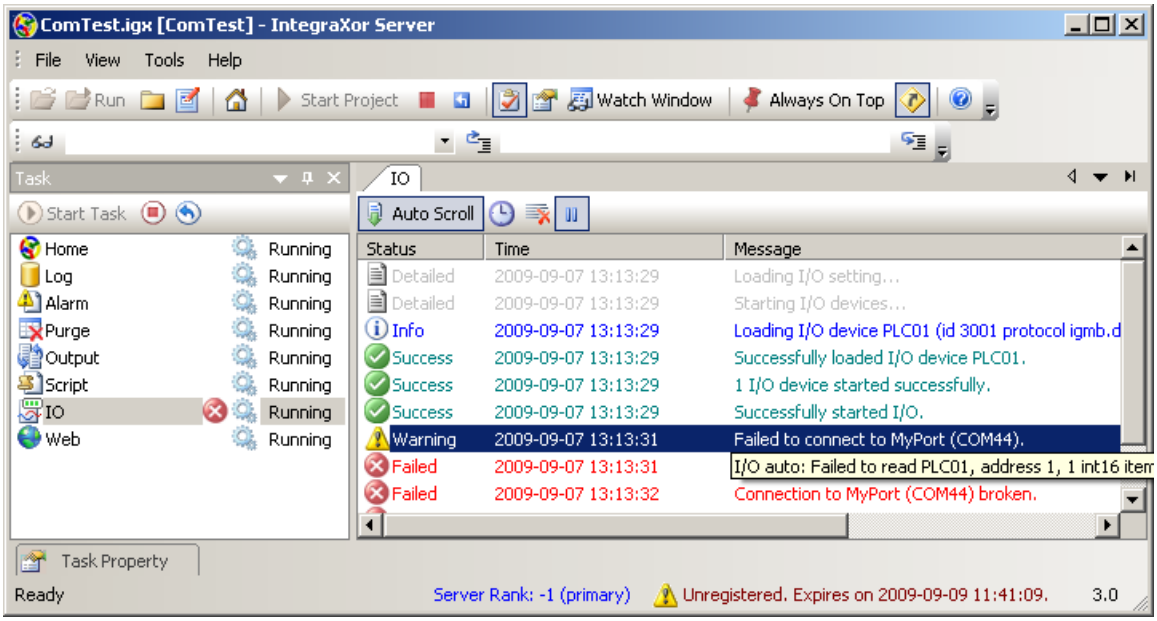
Mismatched of data swapping will not trigger any Modbus communication error but the data polled or sent could be showing non-readable values.

Batch Data Transmission

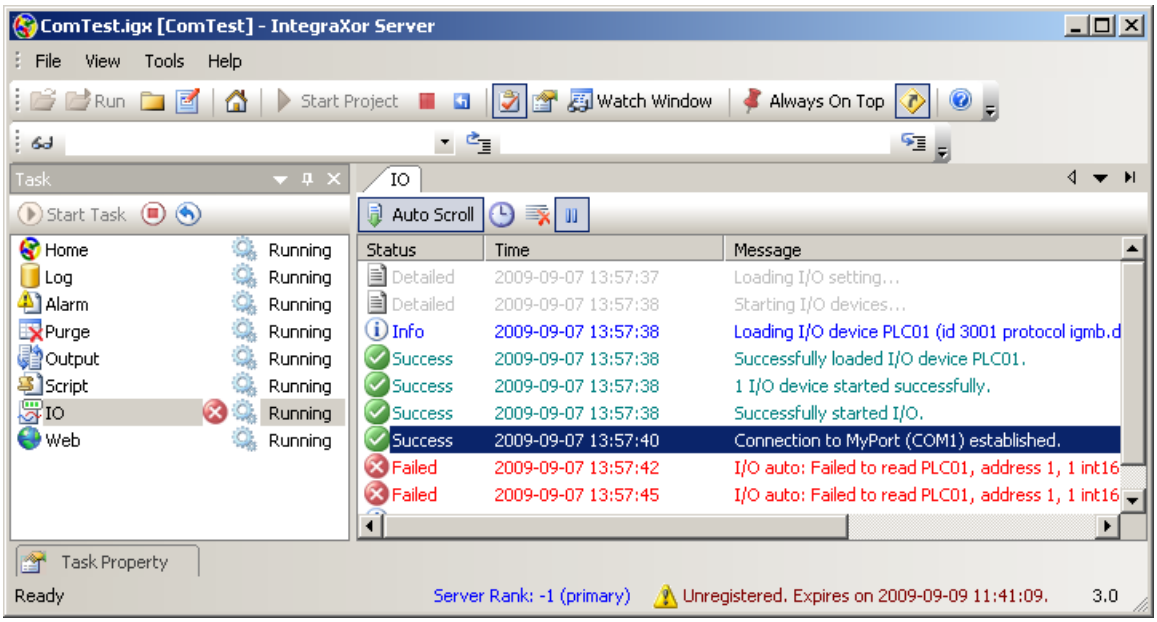
Batch Optimization also being called Block/Contiguous optimization, it is needed when a project has big entry of tags. In fact, almost all Modbus entry requires 'Batch Optimization' regardless of project tag count. Modbus data can be transmitted in batch (one big chunk that sorted contiguously) to speed up the communication. The data address of this batch of data must be in same type and same length. For instance, address 4000, 4001, 4002, 4003 can all be either read or written in one transmission. However, address 10200 (1x) and 40200 (4x) cannot be polled in one go.

9.4. Interpreting Communication Message

1. When a configured port is not available, a warning message will first appear: "Failed to connect to PortName(COMn)." where "PortName" is user specified port name and n is the physical port number. Subsequently Failed messages will keep on generating based on the frequency of the timer.

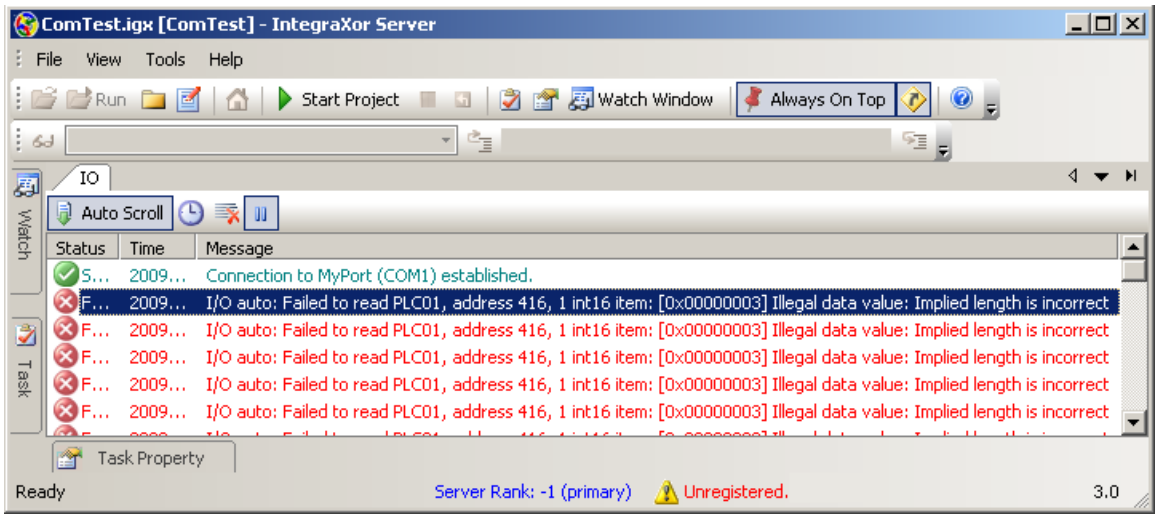

ComTest.igx [ComTest] - IntegraXor Server
File View Tools Help
Run Start Project Watch Window Always On Top
Task
Start Task
Home Running
Log Running
Alarm Running
Purge Running
Output Running
Script Running
IO Running
Web Running
IO
Auto Scroll
Status Time Message
Detailed 2009-09-07 13:13:29 Loading I/O setting...
Detailed 2009-09-07 13:13:29 Starting I/O devices...
Info 2009-09-07 13:13:29 Loading I/O device PLC01 (id 3001 protocol igmb.d
Success 2009-09-07 13:13:29 Successfully loaded I/O device PLC01.
Success 2009-09-07 13:13:29 1 I/O device started successfully.
Success 2009-09-07 13:13:29 Successfully started I/O.
Warning 2009-09-07 13:13:31 Failed to connect to MyPort (COM44).
Failed 2009-09-07 13:13:31 I/O auto: Failed to read PLC01, address 1, 1 int16 item
Failed 2009-09-07 13:13:32 Connection to MyPort (COM44) broken.
Task Property
Ready
Server Rank: -1 (primary)
Unregistered. Expires on 2009-09-09 11:41:09.
3.0
Error messages when specified port is not available.

2. When a configured port can be open successfuly, then `Success` message will be displayed: "Connection to PortName (COMn) established", where PortName is user specified name and n is the physical port number. However, if the cable is not connected or broken, or, the device is not responding which may due to non-matching com port setting, or, wrong node address then the similar `Failed` message will still be keep on generating after the `Success` message.


ComTest.igx [ComTest] - IntegraXor Server
File View Tools Help
Run Start Project Watch Window Always On Top
Task
Start Task
Home Running
Log Running
Alarm Running
Purge Running
Output Running
Script Running
IO Running
Web Running
IO
Auto Scroll
Status Time Message
Detailed 2009-09-07 13:57:37 Loading I/O setting...
Detailed 2009-09-07 13:57:38 Starting I/O devices...
Info 2009-09-07 13:57:38 Loading I/O device PLC01 (id 3001 protocol igmb.d
Success 2009-09-07 13:57:38 Successfully loaded I/O device PLC01.
Success 2009-09-07 13:57:38 1 I/O device started successfully.
Success 2009-09-07 13:57:38 Successfully started I/O.
Success 2009-09-07 13:57:40 Connection to MyPort (COM1) established.
Failed 2009-09-07 13:57:42 I/O auto: Failed to read PLC01, address 1, 1 int16
Failed 2009-09-07 13:57:45 I/O auto: Failed to read PLC01, address 1, 1 int16
Task Property
Ready
Server Rank: -1 (primary)
Unregistered. Expires on 2009-09-09 11:41:09.
3.0
Error messages when cable is disconnected or device not responding.

3. If wrong data type is specified, more detail information will be supplied after the data adddres and data type: `Illegal data value: Implied length is incorrect`. This could well means most of the essential steps are correctly configured, it's confirmed that all comm port settings are matched and node address is correct too. So the only thing left unmatched is the data address. However, one possibility is such that the data address is correct, but the address offset is not tuned correctly. Try to set +1, -1 or 0

into this parameter if the data type is confirmed as matched.


ComTest.igx [ComTest] - IntegraXor Server
File View Tools Help
Run
Start Project
Watch Window
Always On Top
IO
Auto Scroll
Watch
Task
Status Time Message
S... 2009... Connection to MyPort (COM1) established.
F... 2009... I/O auto: Failed to read PLC01, address 416, 1 int16 item: [0x00000003] Illegal data value: Implied length is incorrect
F... 2009... I/O auto: Failed to read PLC01, address 416, 1 int16 item: [0x00000003] Illegal data value: Implied length is incorrect
F... 2009... I/O auto: Failed to read PLC01, address 416, 1 int16 item: [0x00000003] Illegal data value: Implied length is incorrect
F... 2009... I/O auto: Failed to read PLC01, address 416, 1 int16 item: [0x00000003] Illegal data value: Implied length is incorrect
F... 2009... I/O auto: Failed to read PLC01, address 416, 1 int16 item: [0x00000003] Illegal data value: Implied length is incorrect
Task Property
Ready
Server Rank: -1 (primary)
Unregistered.
3.0

Error messages when wrong data type is specified.

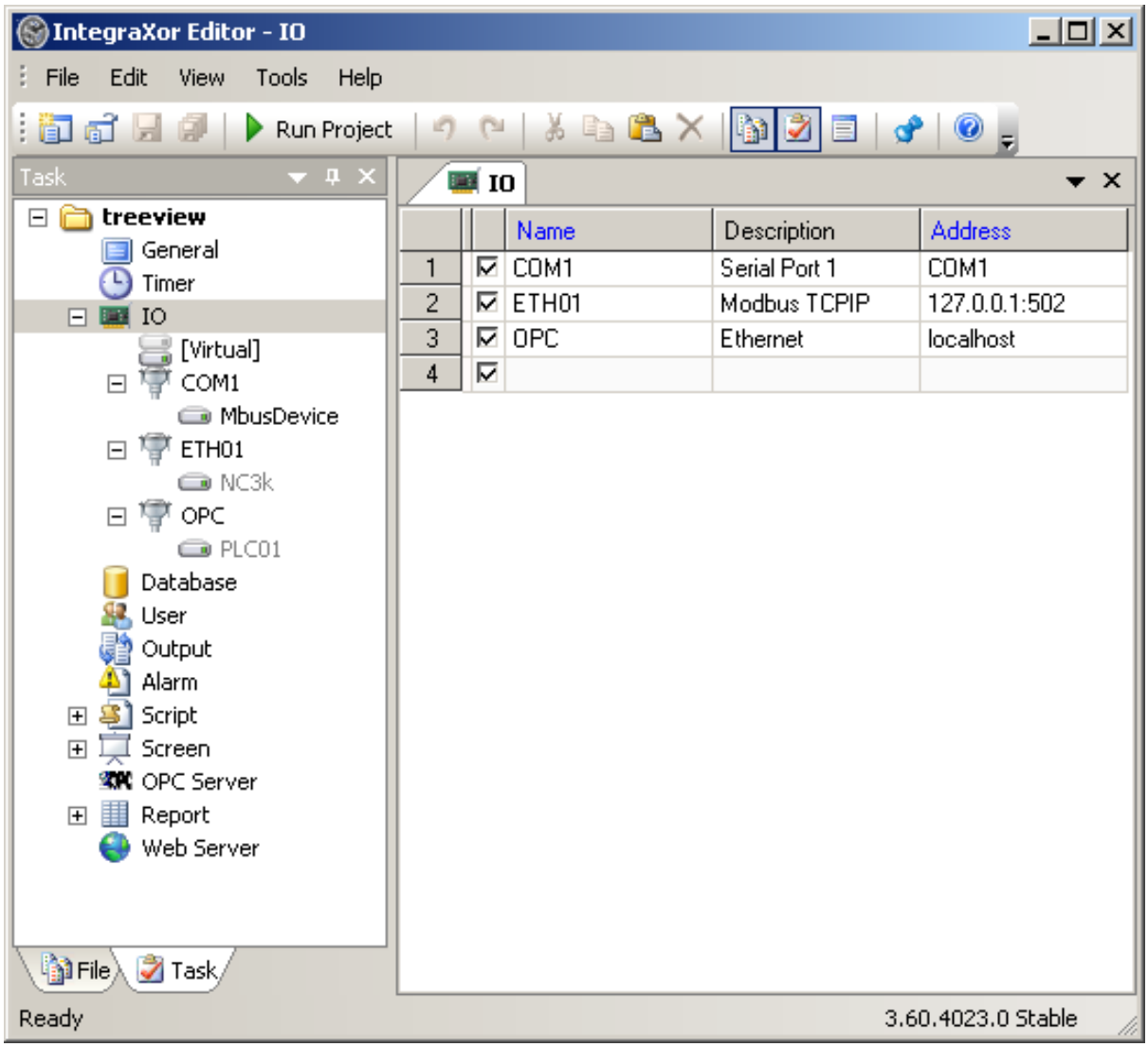
9.5. OPC Troubleshooting for SCADA

OPC configuration for SCADA as client or server won't have much problem as IntegraXor will automatically configure the the registry entries for DCOM. However, the remote OPC client/server may be tricky if the remote OPC client/server itself doesn't automatically configure the necessary entry. Please download and run the registry file [http://www.integraxor.com/developer/tools/opc/remote/opc_dcom.reg] to configure the remote client/server automatically to allow external access.

Chapter 10. Device Driver

$Date:: 2011-01-25 #$

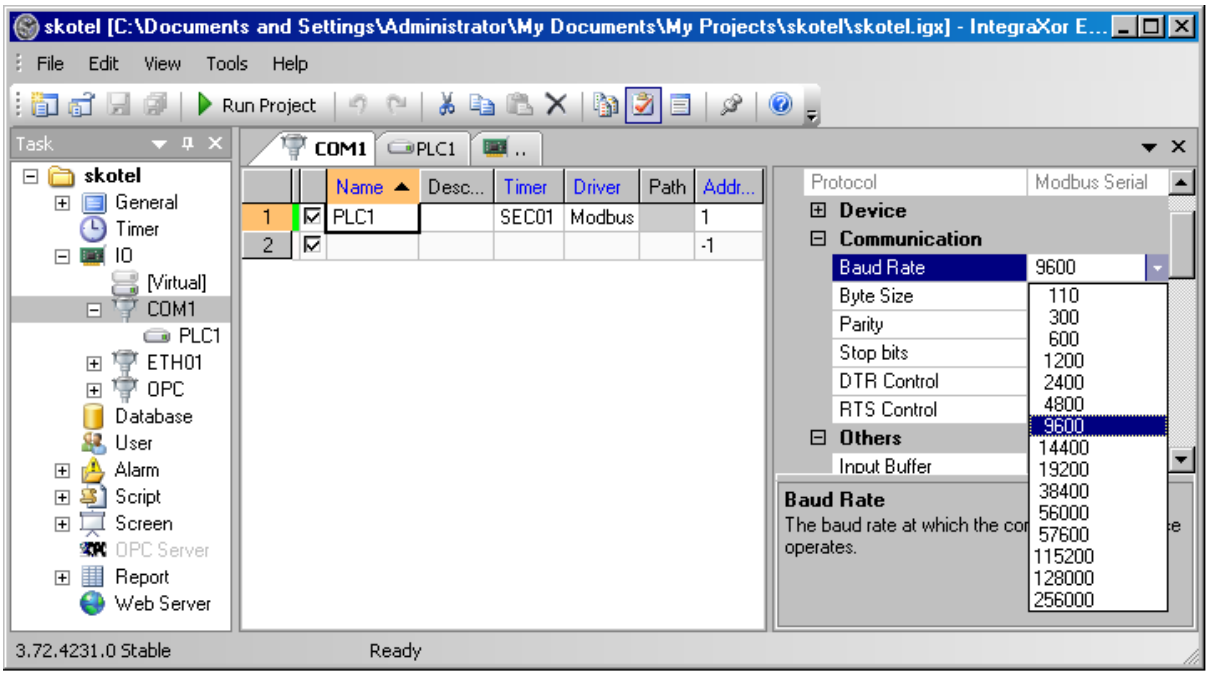
This chapter describes the configuration for all major device drivers. The basic tree view concept for I/O configuration is very consistent and similar through out different protocols. They just need to start by configuring a relevant Port, and followed by the connected Devices. There's no limit of devices to be connected under anyone of the port, which is similar to its network topology.



I/O tree configuration for Ports, Devices and Tags.

10.1. Modbus

Modbus protocol is an very old protocol and has many variants of implementation. IntegraXor supports Modbus for both Serial & Ethernet, i.e. RS232/485/422 & TCPIP. It shall be configured by specifying the type of port and the device configuration will vary accordingly.

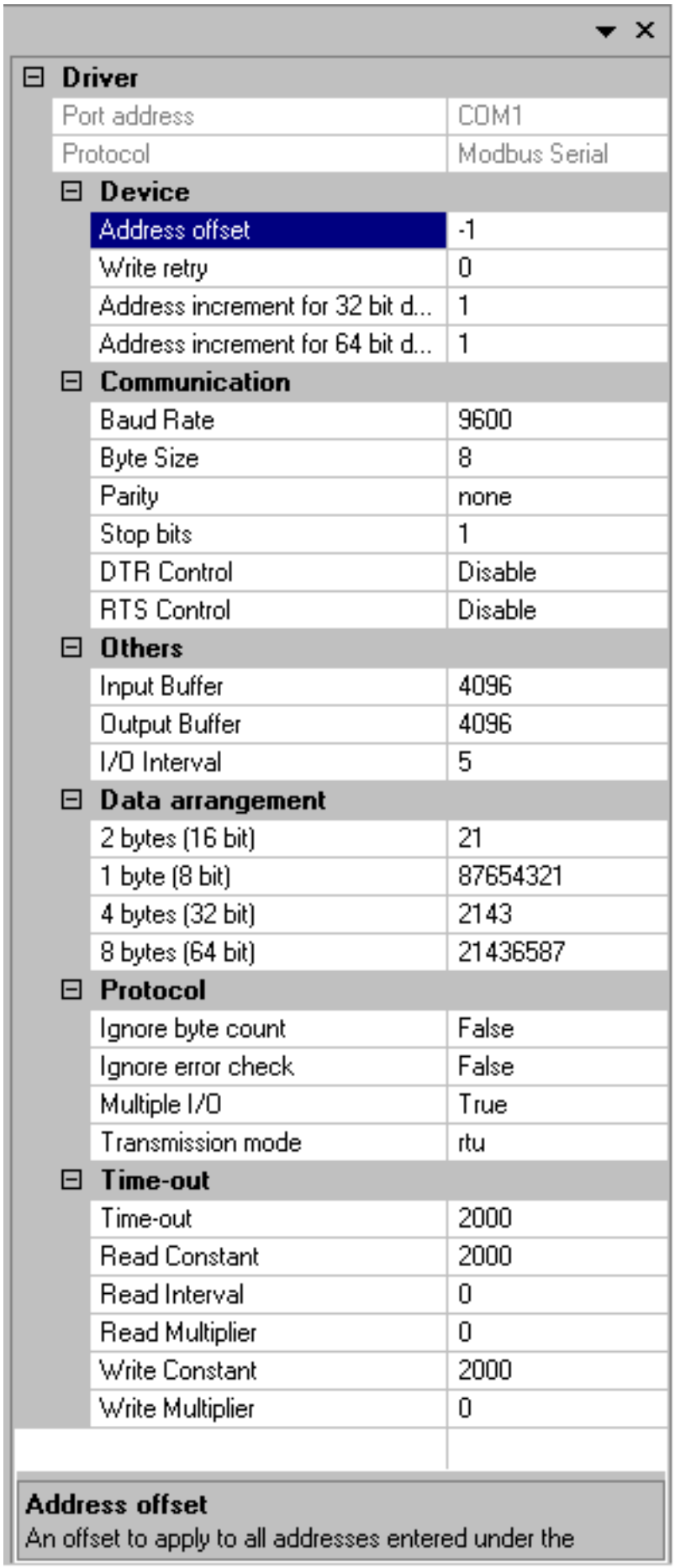


Calling out device communication properties

10.1.1. Port & Device

When Port entry is entered with serial port name of "COM1", "COM2", or subsequent serial port number, the Device properties will be made available for serial port configuration automatically. On the other hand, Modbus TCP/IP device shall be entered with its IP address with port number like "192.168.1.1:502". Note that the trailing ":502" is the standard Ethernet port number for Modbus, it should be the same in most Modbus devices.

Other than the serial communication properties, the essential configurations are the same for both type of devices. Information for each properties item is displayed at the right bottom when each item is clicked or being configured.



Driver	
Port address	COM1
Protocol	Modbus Serial
Device	
Address offset	-1
Write retry	0
Address increment for 32 bit d...	1
Address increment for 64 bit d...	1
Communication	
Baud Rate	9600
Byte Size	8
Parity	none
Stop bits	1
DTR Control	Disable
RTS Control	Disable
Others	
Input Buffer	4096
Output Buffer	4096
I/O Interval	5
Data arrangement	
2 bytes (16 bit)	21
1 byte (8 bit)	87654321
4 bytes (32 bit)	2143
8 bytes (64 bit)	21436587
Protocol	
Ignore byte count	False
Ignore error check	False
Multiple I/O	True
Transmission mode	rtu
Time-out	
Time-out	2000
Read Constant	2000
Read Interval	0
Read Multiplier	0
Write Constant	2000
Write Multiplier	0

Address offset
An offset to apply to all addresses entered under the

Modbus device properties

Device - Address offset	Adjust the address forward or backward if it is mismatched between SCADA and device.
Device - Write retry	Number of retrial upon write error/timeout.
Device - Address increment for 32 bit data	This setting is applipable for Floating point or Long integer data. Some devices like Schneider PLC's 32 bit data constitute of 2 addresses, so the address for floating point data listing will appear to be skipped with either odd numbers or even numbers. In such case, this setting needs to set to value of 2.
Device - Address increment for 64 bit data	This setting is applipable for Floating point or Long integer data. Some devices' 64 bit data constitute of 4 addresses, so the address for floating point data listing shall be incremental by 4. In such case, this setting needs to set to value of 4.
Data arrangement - 2 bytes (16 bit)	Some device may have an integer data swapped. This setting shall allow little endian or big endian data convention.
Data arrangement - 1 byte (8 bit)	This setting shall allow little endian or big endian data convention.
Data arrangement - 4 bytes (32 bit)	This setting is more often need to change as compare to the rest of the data byte size, as it's meant for commonly used float 32 or long 32 data, in performing data swapped display. Normally the setting shall be "2143" or "4321".
Data arrangement - 8 bytes (64 bit)	This setting allows double 64 or long 64 byte swapping just like the rest of the different length data.

10.1.2. Tag

Traditionally there are only 4 types of data appears in Modbus PLC, i.e. Holding Register (4x), Input Register (3x), Output Coil (0x) & Input Coil (1x), whereby Holding Register & Input Register are 16 bit integer wherease Output Coil & Input Coil are Boolean data type. Floating point & Long analogue data in 32 bit or 64 bit can then be stored on multiple Holding or Input Registers. To configure a tag entry for Input Register (3x) or Input Coil (1x) will need to set the Input Tag of Tag Properties to "True", hence they are both read-only tag and the value can't be changed by using `setTag()` function call in Script.

10.2. EtherNet Industrial Protocol

EtherNet/IP™ was developed by Rockwell Automation in late 1990, and handed over to ODVA in 2001 which is now an open industrial protocol.

EtherNet/IP only works over TCP/IP, as such the port must be configured with IP address. Two parameters that need to be configured on device are Routing & Controller Slot.

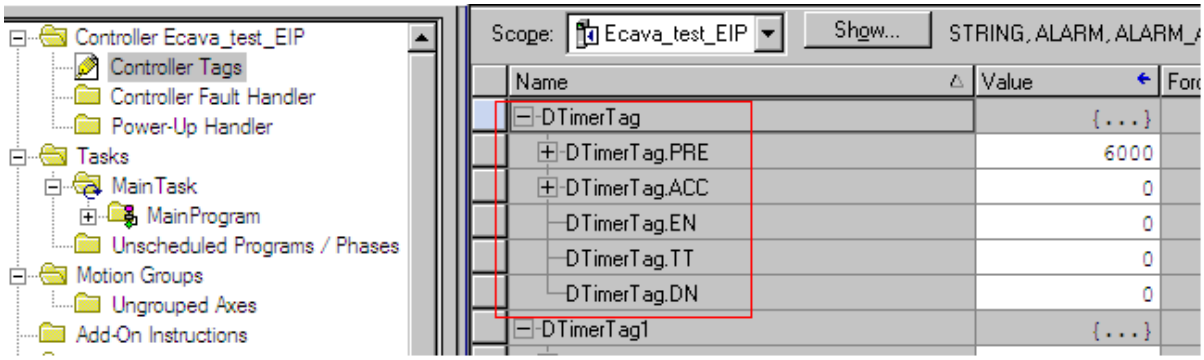
The tag data type must be matched with the PLC according to the following table.

PLC data type	**SCADA data type**
BOOL	boolean

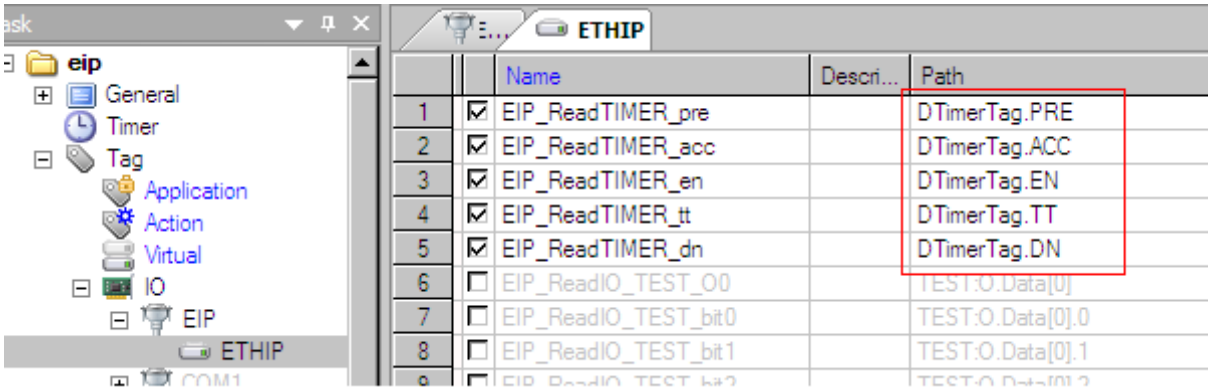
PLC data type	SCADA data type
SINT	int8
INT	int16
DINT	int32
LINT	int64
REAL	real32

EtherNet/IP also supports contiguous polling like Modbus protocol to speed up communication by polling a continous list of addresses. This can be achieved by specifying the batch number for tags with same data type that being arranged contiguously. However auto-batching in Modbus is yet to be supported in EtherNet/IP. Setting value of zero (0) will make the polling separately which is highly discourage in most circumstances.

EtherNet/IP uses string in Path column instead of numerical value in Address column for tag data address. Fill in the Path entry with the device actual tag I/O addressing format. The path string must match with the tag name at Controller Tags, in RSLogix5000 for instance.



Configuring I/O in PLC



Configuring EtherNet/IP tags in SCADA

Note

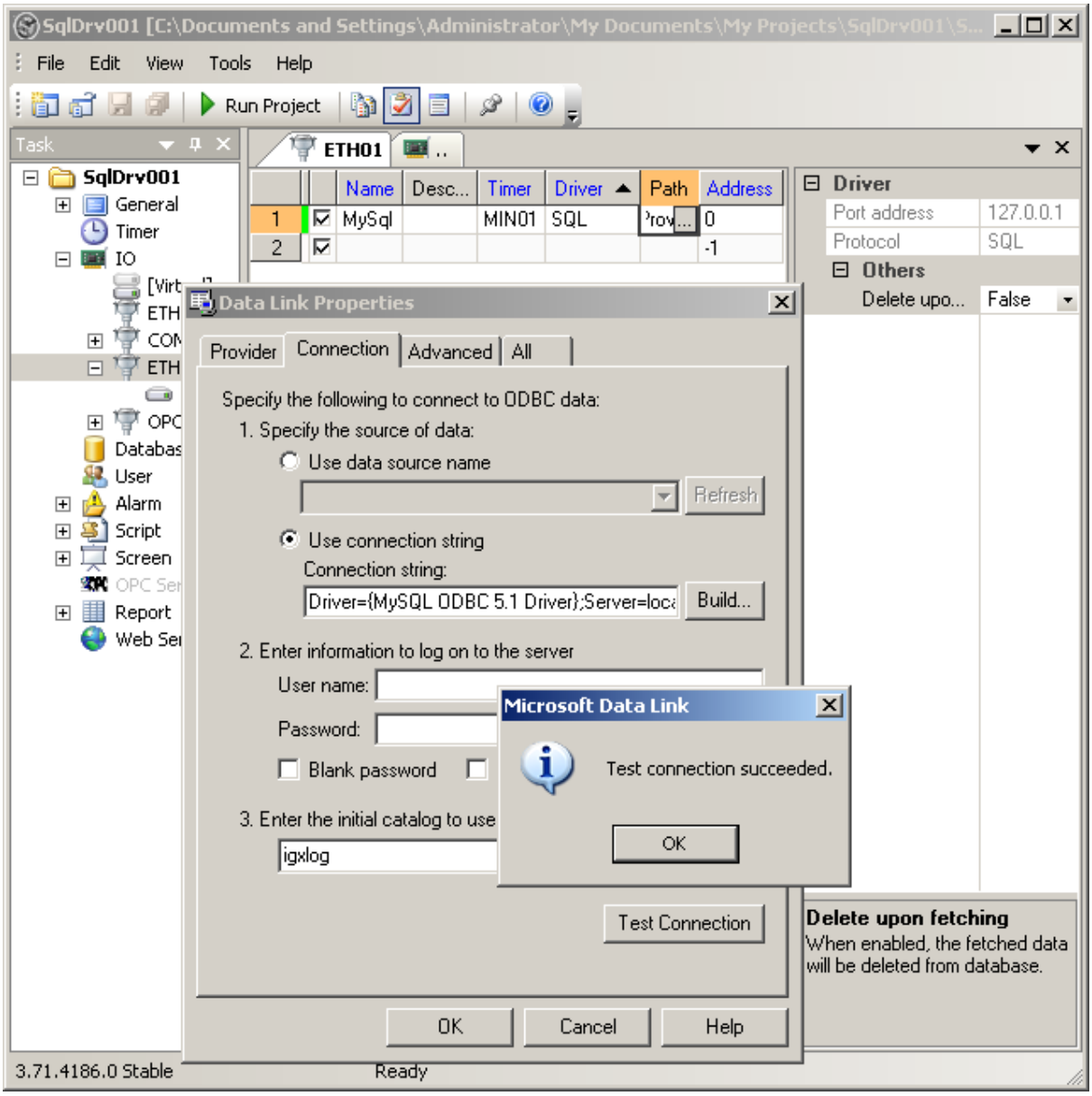
Turn Input tag attribute to true for Read-Only input tag.

10.3. SQL driver for database integration

SQL Driver is designed to communicate with External/Additional Database System that has its own schema and data source. It can establish the link with any brand of SQL Database so long it supports ODBC.

10.3.1. Port & Device

Using SQL Driver is pretty straightforward, first is to define a Ethernet port that having the IP address point to where the database is running. Secondly is to add an entry for SQL driver and choose its polling rate accordingly. Third step is different from other drivers whereby the Path field shall be entered as Database Connection String. For instance, user may enter the following for MySQL database: `"Provider=MSDASQL.1;Persist Security Info=False;Extended Properties="Driver={MySQL ODBC 5.1 Driver};Server=localhost;Database=igxlog;User=root;Option=3";Initial Catalog=igxlog"`. The Address field is not used for device entry, enter 0 if the version used requires this field to be filled.



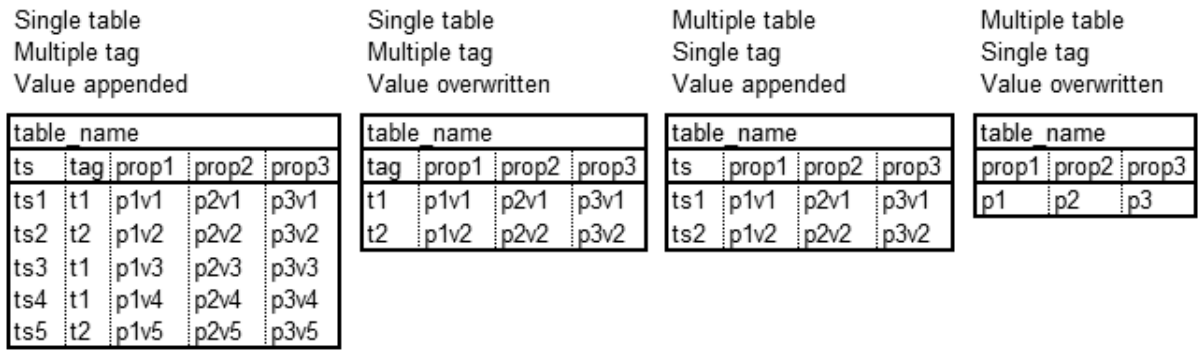
Configuring Port for SQL driver

Note that user could choose to delete the data upon fetching by enabling this option on the far right.

10.3.2. Tag

Similar like device path, Address field for tag entry also not used but the Path field requires to be entered with a

specific syntax for associating with a Table column within a Database Schema. There are several possible database schema as shown at the following diagram.



Single table
Multiple tag
Value appended

table_name				
ts	tag	prop1	prop2	prop3
ts1	t1	p1v1	p2v1	p3v1
ts2	t2	p1v2	p2v2	p3v2
ts3	t1	p1v3	p2v3	p3v3
ts4	t1	p1v4	p2v4	p3v4
ts5	t2	p1v5	p2v5	p3v5

Single table
Multiple tag
Value overwritten

table_name			
tag	prop1	prop2	prop3
t1	p1v1	p2v1	p3v1
t2	p1v2	p2v2	p3v2

Multiple table
Single tag
Value appended

table_name			
ts	prop1	prop2	prop3
ts1	p1v1	p2v1	p3v1
ts2	p1v2	p2v2	p3v2

Multiple table
Single tag
Value overwritten

table_name		
prop1	prop2	prop3
p1	p2	p3

Various possible database schema for interaction.

The syntax of the parameter is straightforward as per the following format. It was designed to be as close as per the SQL statement, yet simple enough for beginner to pickup, and most importantly it works across different database:

```
database_table_name/function_name:data_column_name[sql_where_condition
]@time_stamp_column_name
```

The main structure of the syntax that consists of mandatory `database_table_name` and `data_column_name` is pretty self explanary, except the optional `function_name:`, whereby user may perform algorithms like `min:`, `max:`, `avg:` & `count:`. Note that `count:` will return numbers of row instead of the value of the specified column.

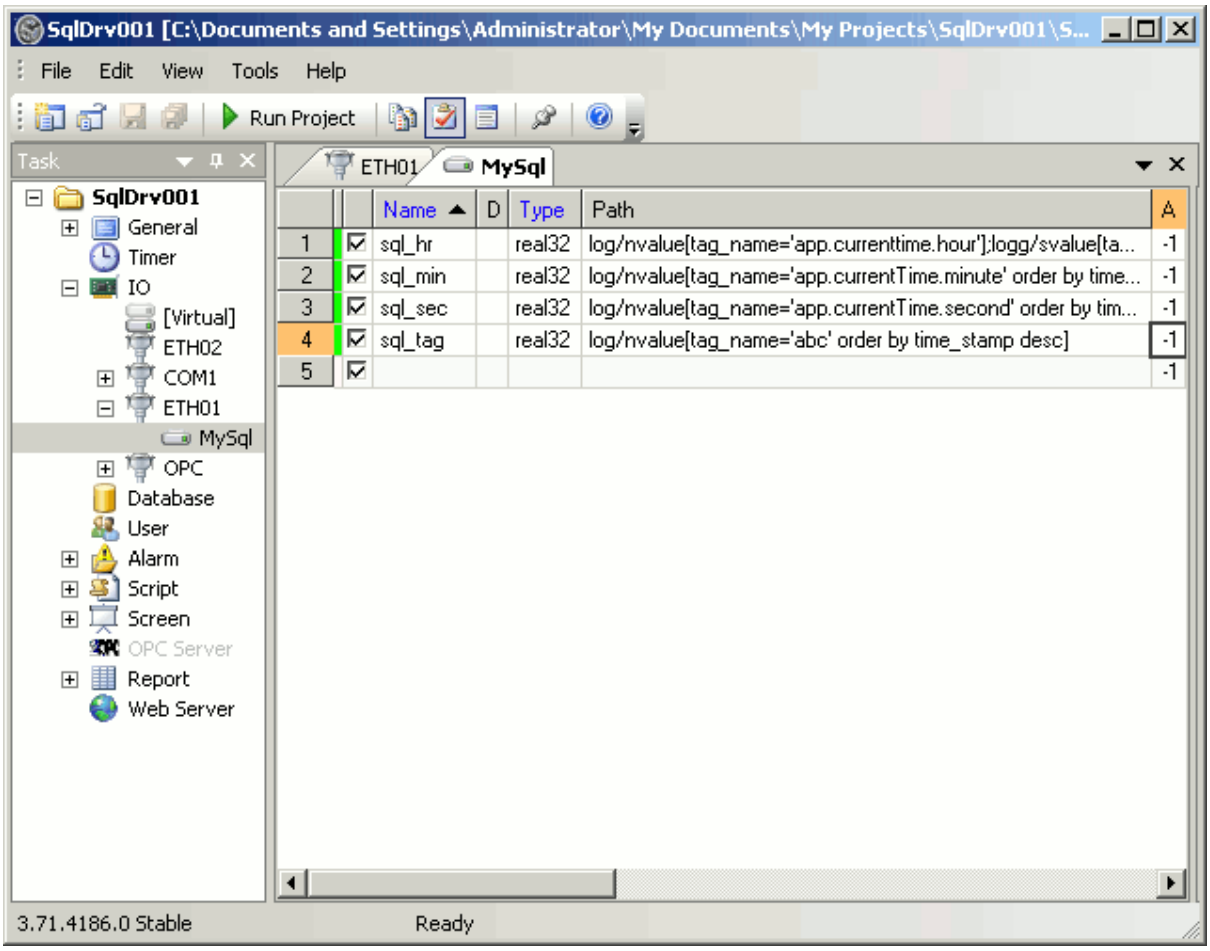
Another optional item that need to be written in square bracket `sql_where_condition` can be entered with any formation that compliant to SQL statement, or further trailed with `order by` sorting method so the top row will be associated to the targetted tag value.

The last optional portion of the syntax is `time_stamp_column_name`, this option allow user to associate external time stamp into IntegraXor live time stamp. This mean, when the tag is used into Alarm Task or Trending Task, the external associated time stamp will be used instead of the actual SCADA time stamp upon receiving the data.

In any case that when the read-from and write-to database table is different, user may insert a semicolon as separator and then append the same syntax to the end for the write operation. Below is an sample syntax when read & write are of different table with different conditions.

```
log1/read_fr_column[tag_name='tag'        order        by        time_stamp
desc];log2/write_to_column[data is not NULL]
```

Project Editor is designed for both mouse and keyboard oriented users. User may enter all syntax directly, copy & paste the content from one field to another or click on the browse button [...] on the right of the syntax entry cell. A syntax builder will pop up to help user in entering the relevant parameter.



Entering SQL driver syntax via composer

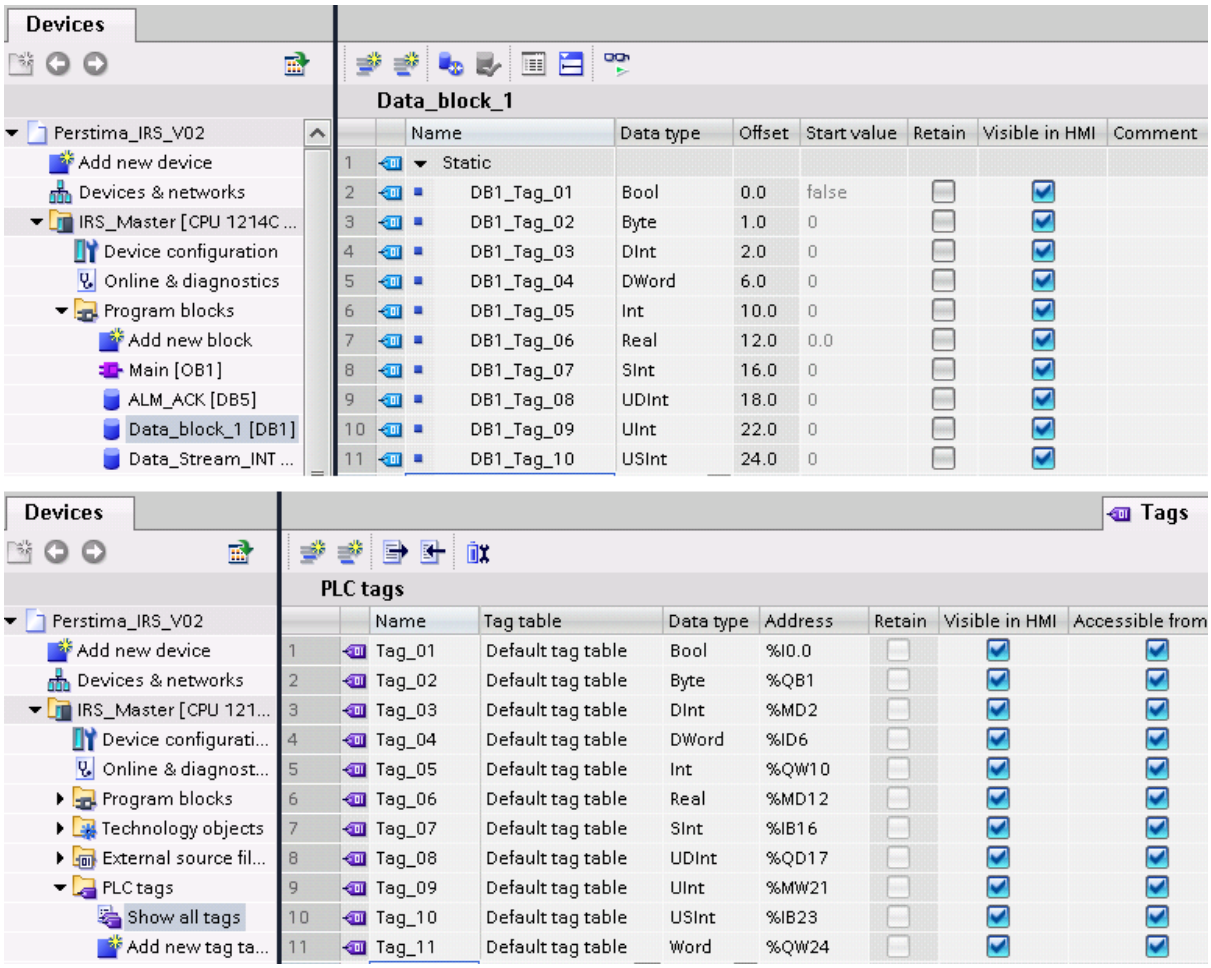
Note

Be aware that there's no verification of syntax or whatsoever fool-proof checking before the database operation is performed, user must perform the test before deployment to site machine.

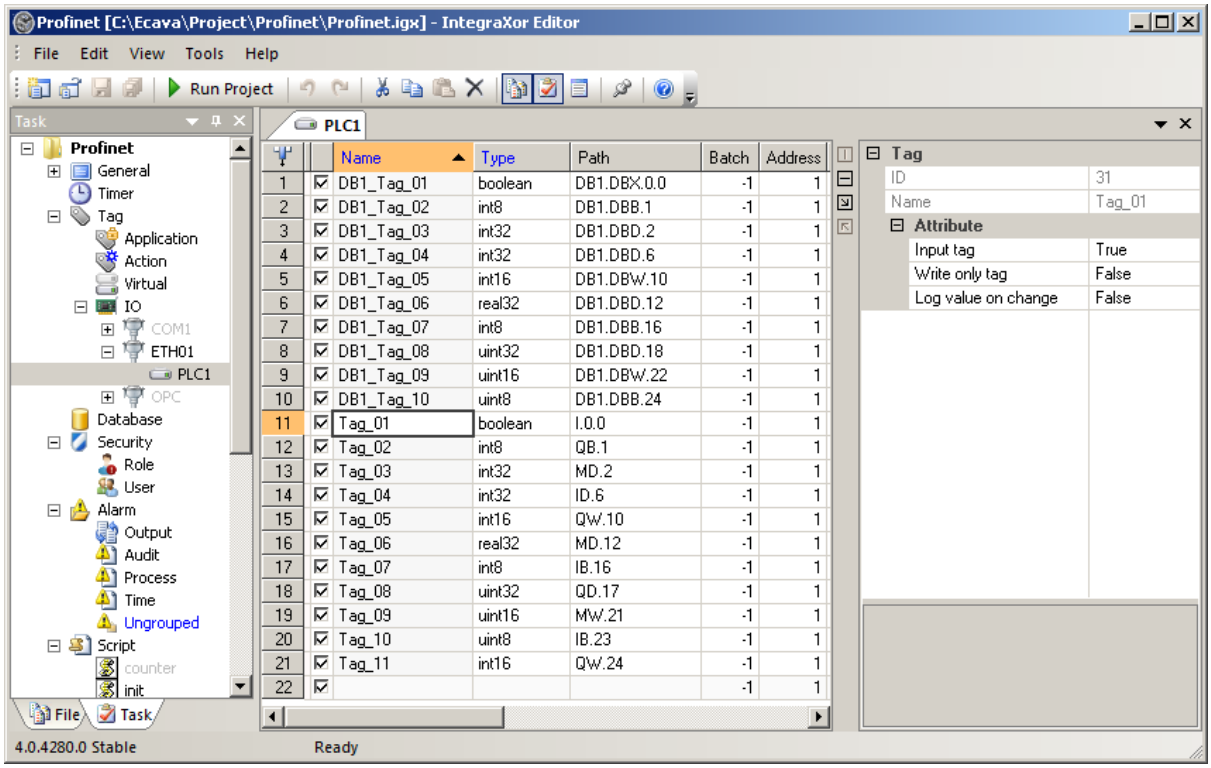
10.4. Profinet

PROFINET™ is the open industrial Ethernet standard of PROFIBUS & PROFINET International (PI) for automation. PROFINET™ uses TCP/IP and IT standards, and is, in effect, real-time Ethernet, as such the port must be configured with IP address. Two parameters that need to be configured on device are CPU Slot & Station Name.

PROFINET™ uses string in Path column instead of numerical value in Address column for tag data address. Fill in the Path entry with the device actual tag I/O addressing format. The path string must match with the tag name at Siemens S7-1200 Tags, in TIA Portal for instance.


Devices
Perstima_IRS_V02
Add new device
Devices & networks
IRS_Master [CPU 1214C ...
Device configuration
Online & diagnostics
Program blocks
Add new block
Main [OB1]
ALM_ACK [DB5]
Data_block_1 [DB1]
Data_Stream_INT ...
Data_block_1
Name | Data type | Offset | Start value | Retain | Visible in HMI | Comment
1 Static
2 DB1_Tag_01 Bool 0.0 false
3 DB1_Tag_02 Byte 1.0 0
4 DB1_Tag_03 DInt 2.0 0
5 DB1_Tag_04 DWord 6.0 0
6 DB1_Tag_05 Int 10.0 0
7 DB1_Tag_06 Real 12.0 0.0
8 DB1_Tag_07 SInt 16.0 0
9 DB1_Tag_08 UDInt 18.0 0
10 DB1_Tag_09 UInt 22.0 0
11 DB1_Tag_10 USInt 24.0 0
Devices
Tags
PLC tags
Perstima_IRS_V02
Add new device
Devices & networks
IRS_Master [CPU 121...
Device configurati...
Online & diagnost...
Program blocks
Technology objects
External source fil...
PLC tags
Show all tags
Add new tag ta...
Name | Tag table | Data type | Address | Retain | Visible in HMI | Accessible from
1 Tag_01 Default tag table Bool %I0.0
2 Tag_02 Default tag table Byte %QB1
3 Tag_03 Default tag table DInt %MD2
4 Tag_04 Default tag table DWord %ID6
5 Tag_05 Default tag table Int %QW10
6 Tag_06 Default tag table Real %MD12
7 Tag_07 Default tag table SInt %IB16
8 Tag_08 Default tag table UDInt %QD17
9 Tag_09 Default tag table UInt %MW21
10 Tag_10 Default tag table USInt %IB23
11 Tag_11 Default tag table Word %QW24

Configured I/O in TIA Portal



Configuring PROFINET tags in IntegraXor

As we can see from example screenshot above, the tag data type must be matched with the PLC according to the following table.

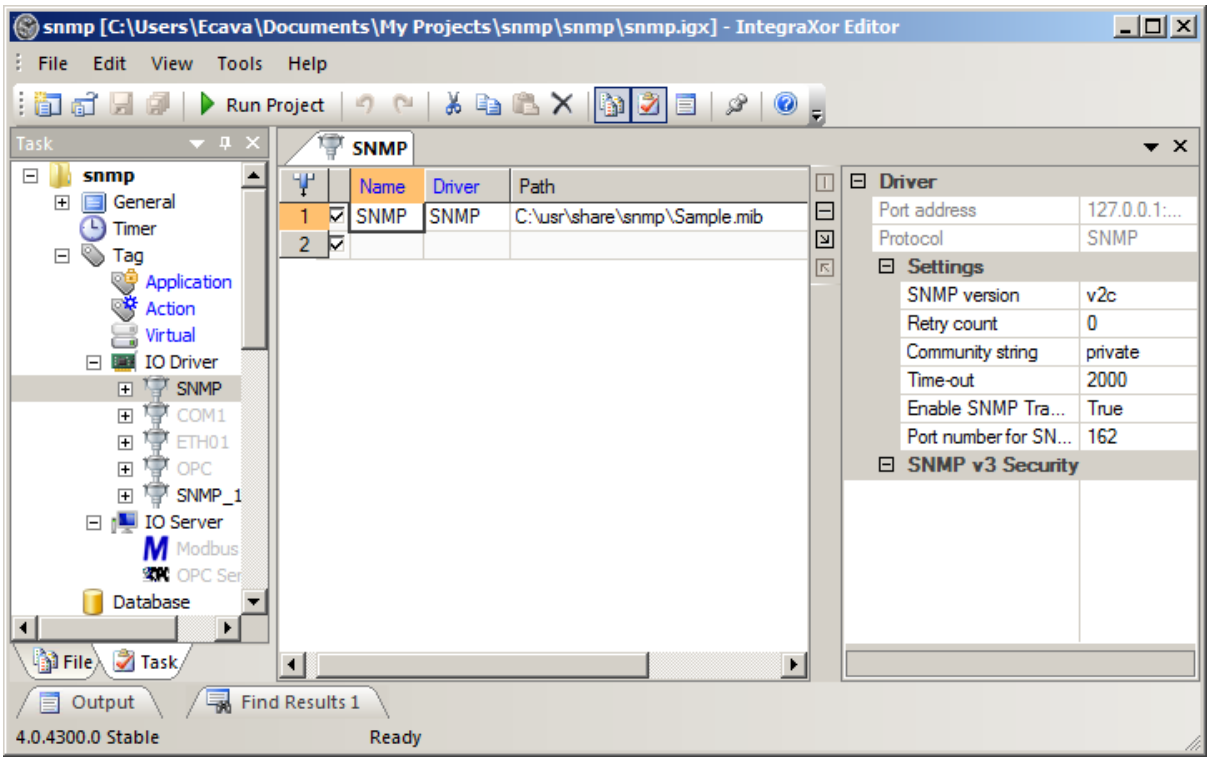
PLC data type	IntegraXor data type
Bool	boolean
Byte / SINT	int8
Int / Word	int16
DInt / Dword	int32
USInt	uint8
UInt	uint16
UDInt	uint32
Real	real32

Note

Turn Input tag attribute to True for Read-Only input tag.

10.5. SNMP

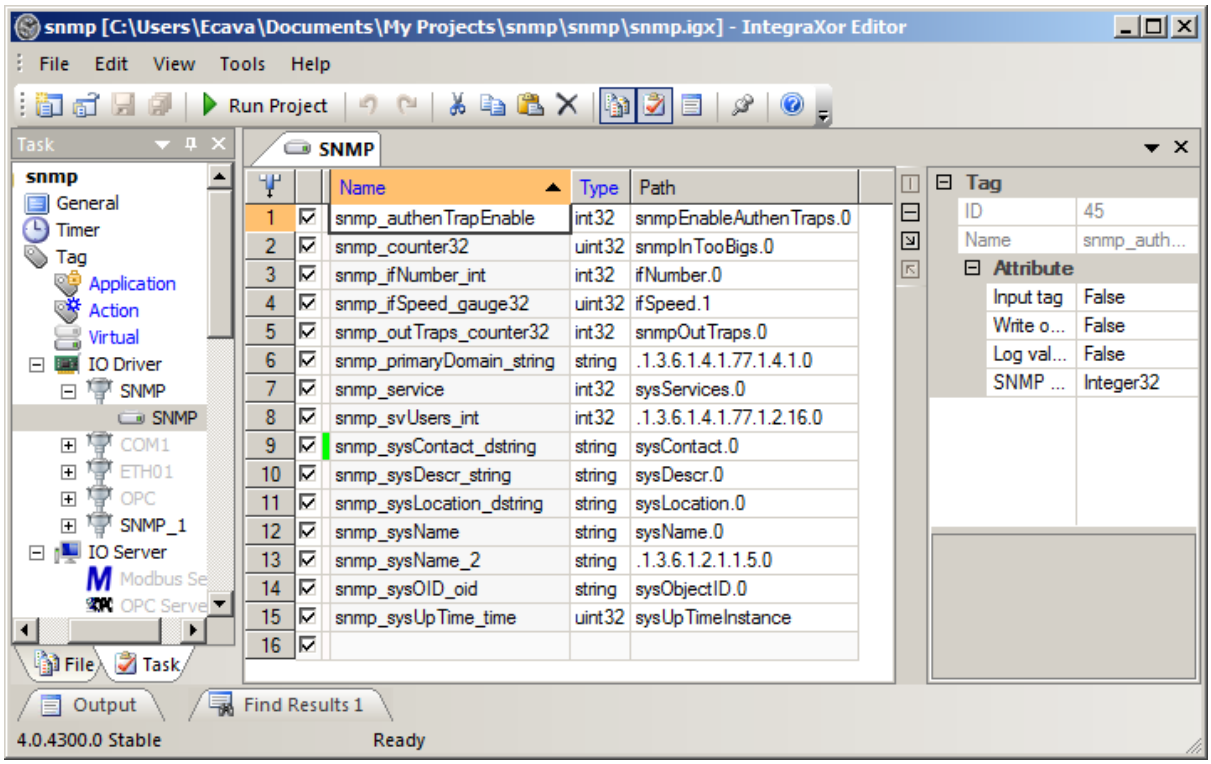
SNMP (Simple Network Management protocol) is the Internet standard protocol to manage network devices. The port is configured with IP address and port number. The default port number is 161. The device settings are SNMP version, Retry count, Community string, Time-out, Enable SNMP Trap/Inform and Port number for SNMP Trap. The path column is configured with the full path of the MIB (Management Information Base) file.



Configuring SNMP device in IntegraXor

The tag setting that needs to be configured is the SNMP data type. The path column is configured with the OID

(Object Identifier) of the tag.



Configuring SNMP tag in IntegraXor

As we can see from example screenshot above, the tag data type must be matched with the SNMP data type according to the following table.

SNMP data type	IntegraXor data type
BITS	string
Counter32	uint32
Counter64	uint64
Gauge32	uint32
INTEGER	int32
Integer32	int32
IpAddress	string
OBJECT IDENTIFIER	string
OCTET STRING	string
Opaque	string
Time Ticks	uint32
Unsigned32	uint32

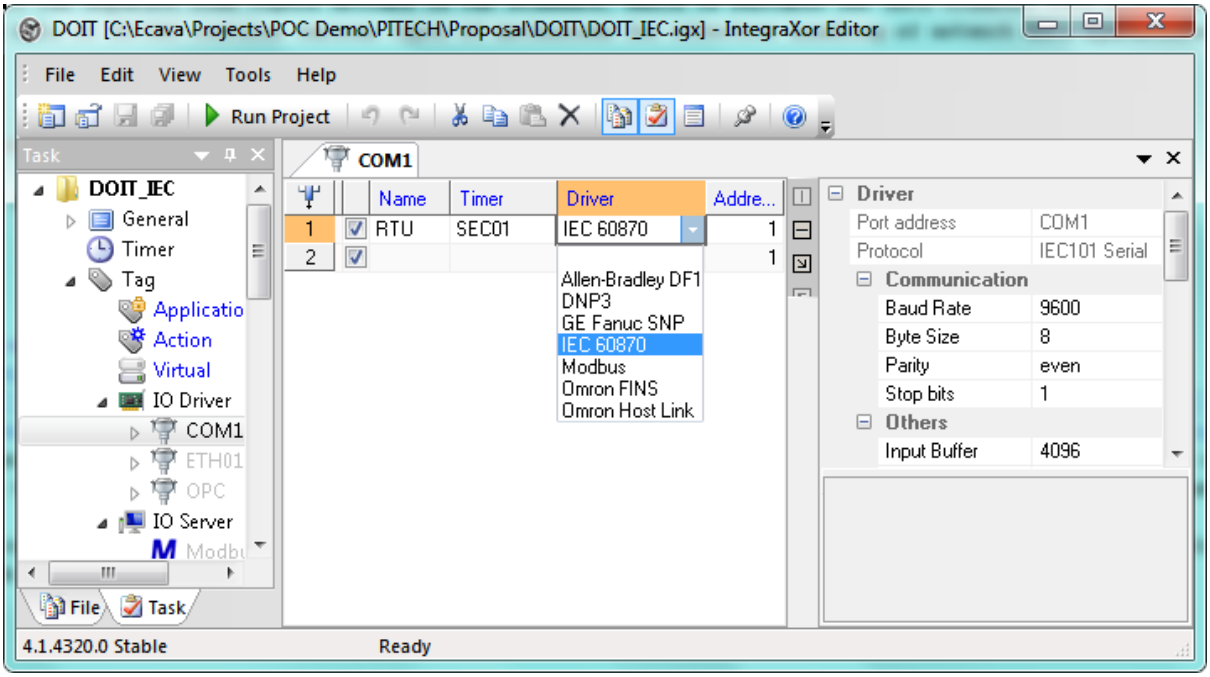
Note

Turn Input tag attribute to True for Read-Only input tag.

10.6. IEC60870-5-101/104

IEC 60870-5-101 [IEC101] is a standard for power system monitoring, control and associated communications for telecontrol, teleprotection, and associated telecommunications for electric power systems. This is completely compatible with IEC 60870-5-1 to IEC 60870-5-5 standards and uses standard asynchronous serial tele-control channel interface between DTE and DCE. The standard is suitable for multiple configurations like point-to-point, star, mutidropped etc.

IEC 60870-5-104 (IEC 104) protocol is an extension of IEC 101 protocol with the changes in transport, network, link and physical layer services to suit the complete network access. The standard uses an open TCP/IP interface to network to have connectivity to the LAN (Local Area Network) and routers with different facility (ISDN, X.25, Frame relay etc.) can be used to connect to the WAN (Wide Area Network). Application layer of IEC 104 is preserved same as that of IEC 101 with some of the data types and facilities not used. There are two separate link layers defined in the standard, which is suitable for data transfer over Ethernet and serial line (PPP - Point-to-Point Protocol). The control field data of IEC104 contains various types of mechanisms for effective handling of network data synchronization.

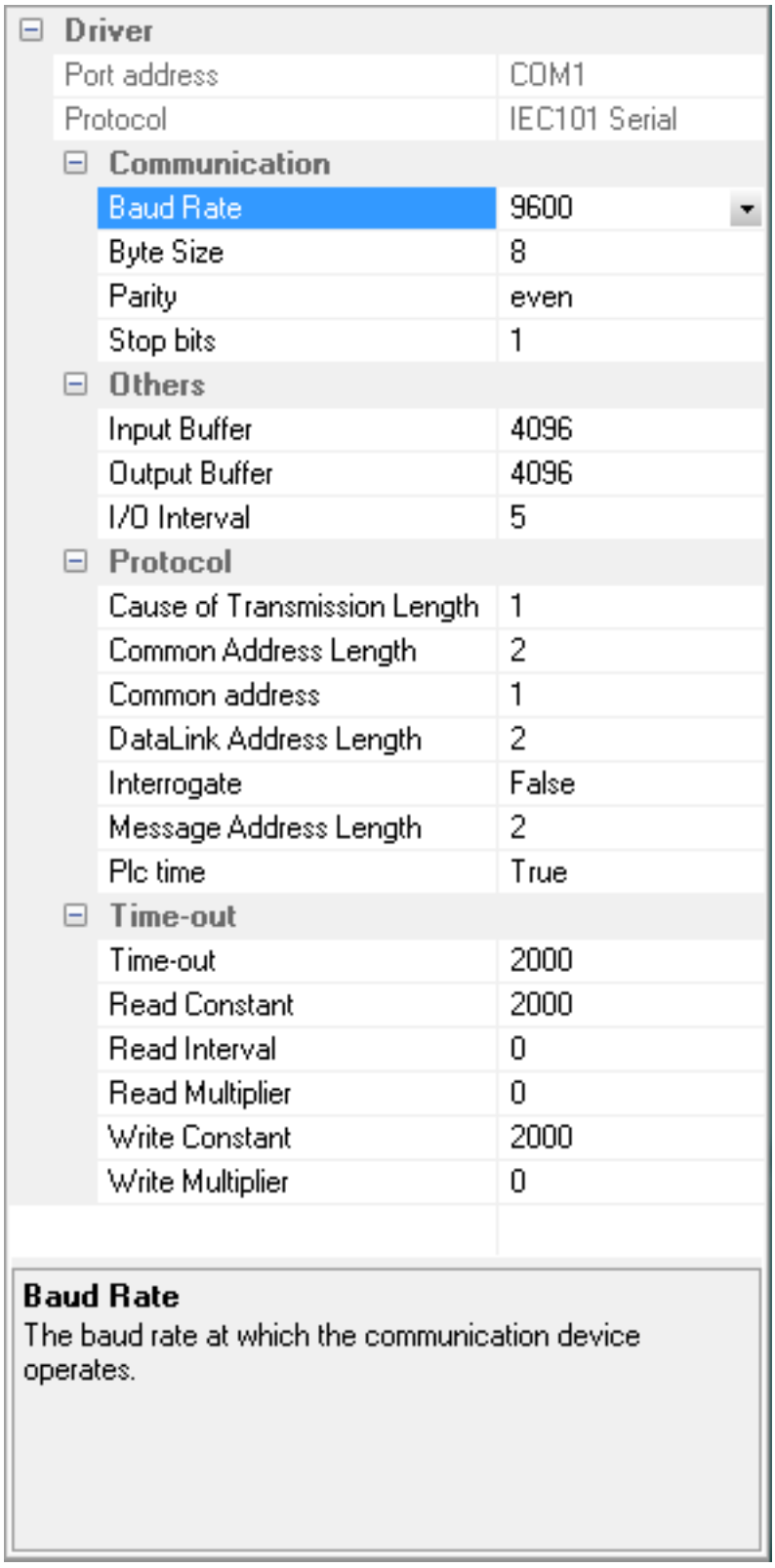


Calling out device communication properties

10.6.1. Port & Device

When Port entry is entered with serial port name of "COM1", "COM2", or subsequent serial port number, the Device properties will be made available for serial port configuration automatically. On the other hand, IEC104 device shall be entered with its IP address with port number like "192.168.1.102".

Other than the serial communication properties, the essential configurations are the same for both type of devices. Information for each properties item is displayed at the right bottom when each item is clicked or being configured.



IEC device properties

Protocol - Cause of Transmission Length Length of the cause of transmission.

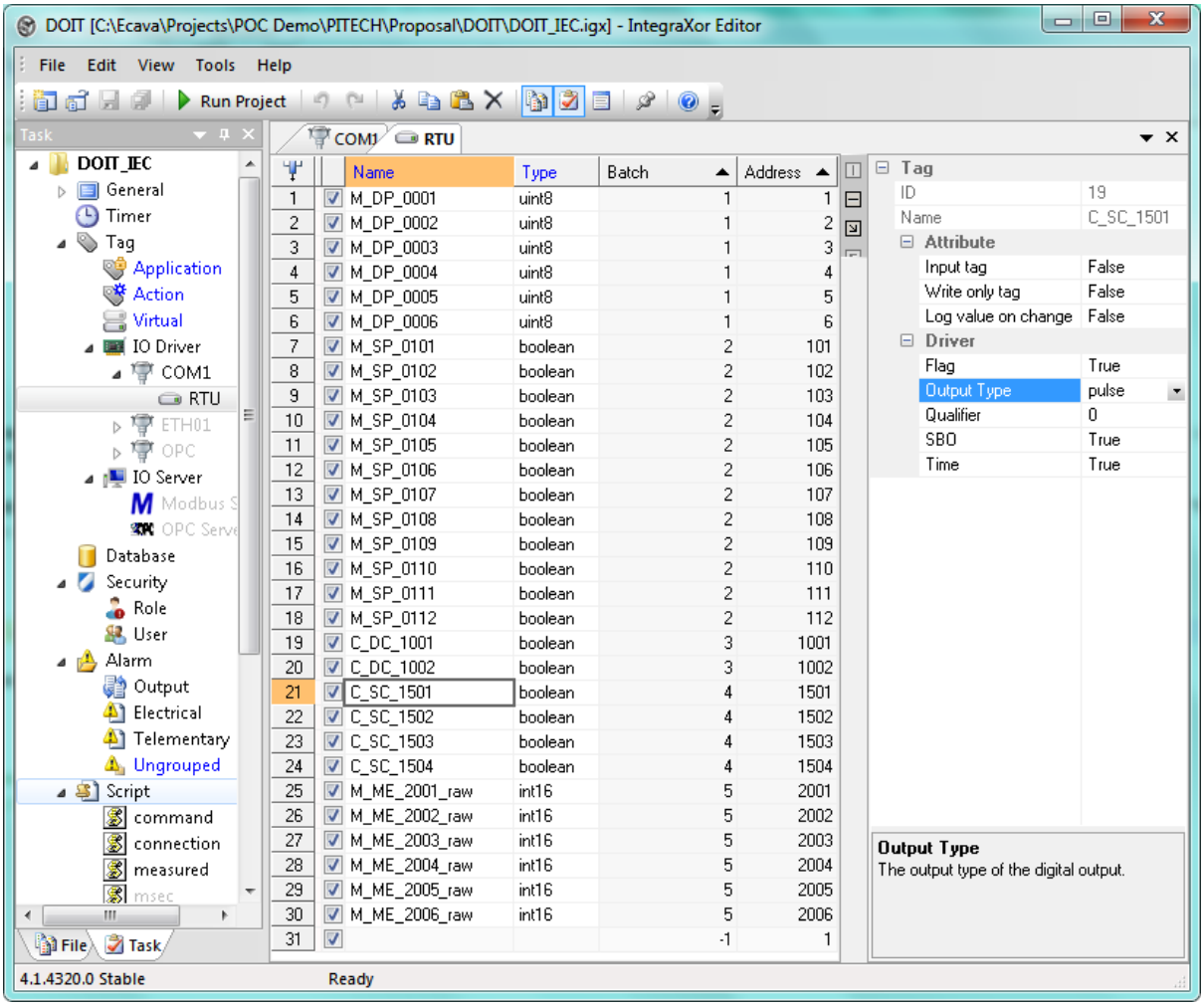
Protocol - Common Address Length Length of the common address.

Protocol - Common Address Common address of the ASDU.

Protocol - DataLink Address Length	Length of the datalink address.
Protocol - Interrogate	Interrogate request when starting.
Protocol - Message Address Length	Length of the message address.
Protocol - PLC Time	True indicates using Outstation time; False indicates using PC local time.

10.6.2. Tag

There are only 5 types of data support in IEC60870, Double Point (DP), Single Point (SP), Measure Value (ME), Double Command (DC) and Single Command (SC) whereby Measure Value is 16 bit integer, Single Point, Double Command and Single Command are Boolean data type and Double Point is unsigned 8 bit integer. To configure a tag entry for Double Point, Single Point or Measure Value will need to set the Input Tag of Tag Properties to "True", hence they are both read-only tag and the value can't be changed by using `setTag()` function call in Script.



IEC Tag Configuration

Chapter 11. Multiple Redundancy

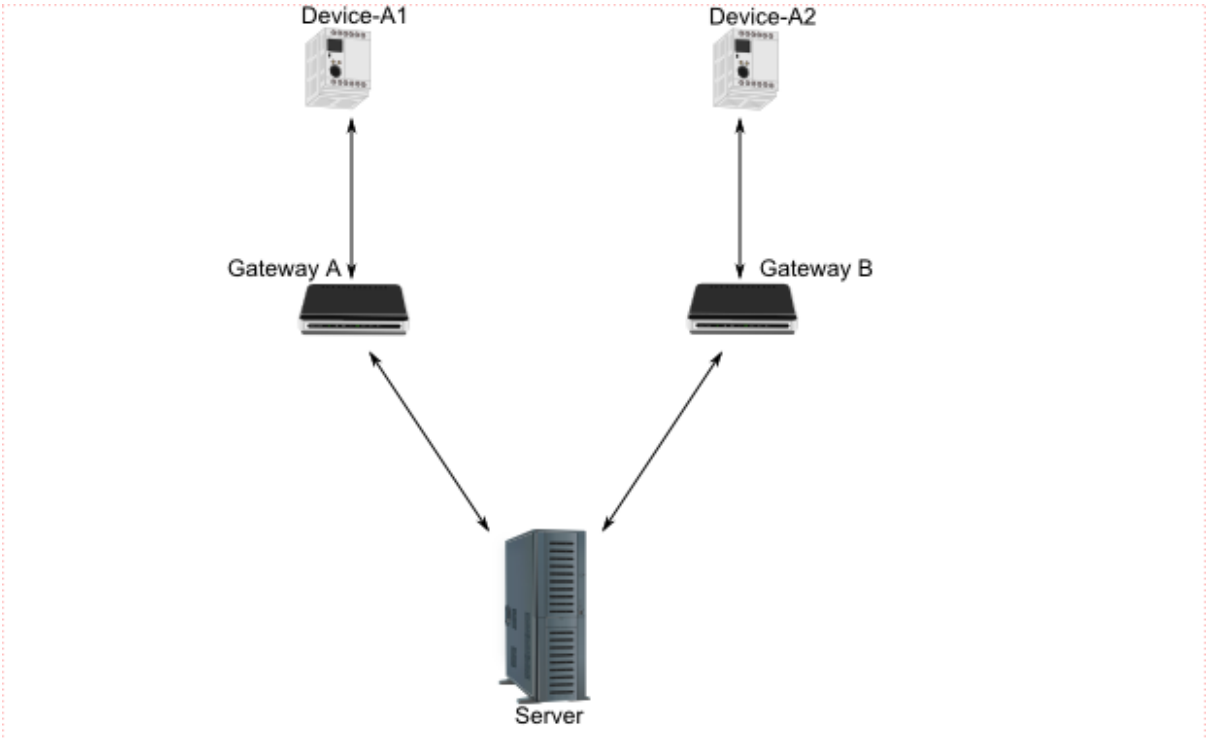
IntegraXor supports dual levels of redundancy, one is Network Level Redundancy and another one is System Level Redundancy. Both of them can either run concurrently or independently and are totally versatile and scalable, which means no limitation for the numbers of redundancy.

On licensing aspect, Network Level Redundancy is a built-in feature that readily available whereas System Level Redundancy will require additional module to run. On the other hand, although Network Level Redundancy allows a tag to be configured once instead of multiple times, the tag count will still based on the numbers of channel being configured. For instance, if one redundant device has 100 tags, and 3 channels has been configured for this device, then the I/O tag count will be 300 tags.

11.1. Multiple Network Redundancy

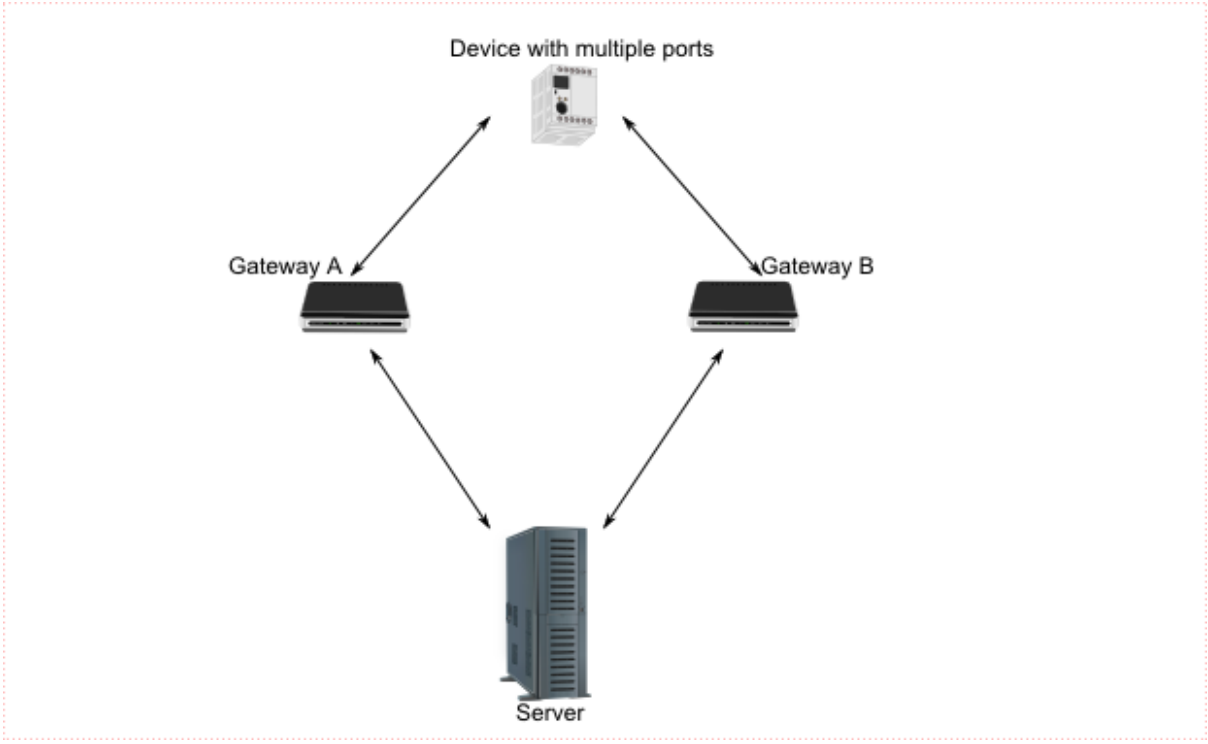
Multiple Network Redundancy enhanced a device/network reliability & efficiency without needing to consider additional cost. It can be implemented on any devices so long the communication driver is available. The redundancy can be achieved on physical device level or solely on network level when a device with multiple ports is available. IntegraXor Network redundancy will greatly simplify and speedup the configuration and programming work on Project Editor, whatever need to be configured multiple times will be reduced to only once.

Only one tag will exist in system layer although the tag value can be polled from different channel or devices when it's configured under the same series of device array. This mean user do not need to bother which channel is available but to use the single tag as if it's from one device/channel.



A configuration that enabled physical device redundancy

In this configuration, two identical physical device is needed. They must perform the same function so IntegraXor will virtually treat it as only one device. Only one set of tags needed to be entered, and the server will poll from any available device at any one time.



A Network redundancy that solely improves the redundant link

This configuration requires less investment on hardware than the previous configuration as it only improves the network links. The pre-requisite for this configuration is the device must have at least 2 or more communication ports, so that the physical cable can be routed via different channel, and the server will only poll via one channel at any one time.

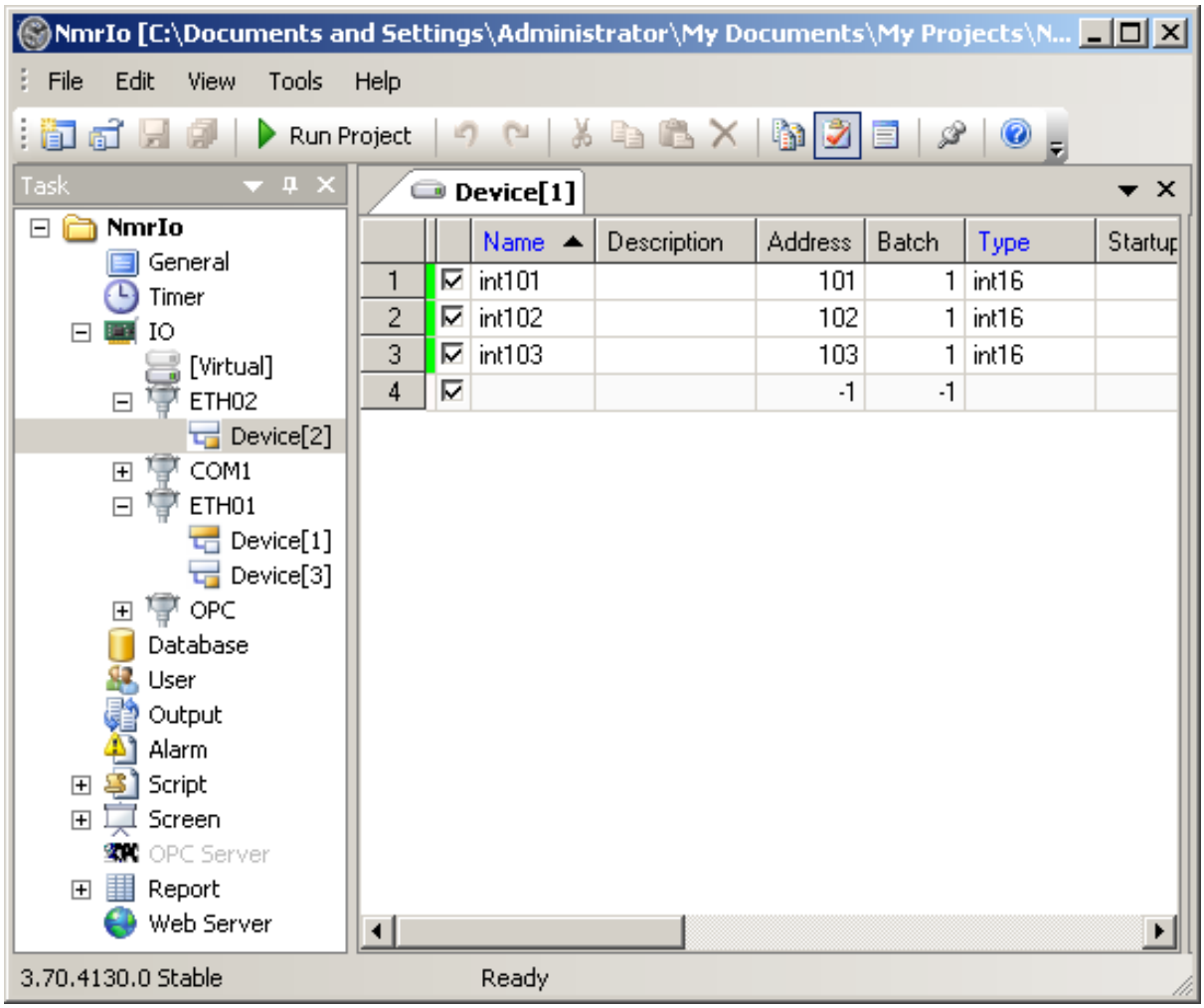
Tip

Not only the numbers of channel is unlimited, but even the protocol can be totally different! This is very useful when some devices provide multiple communication ports but with different protocols, or different physical layer like RS485 can be mixed with RS232 in IntegraXor's configuration.

11.2. Network Redundancy Configuration

Configuring a redundant device/network is fairly straightforward, user only needs to add array symbol `[n]` at the end of the device name to indicate an array of devices shall be mirroring each other or is under the same channel. The array number must always start by number 1, for instance `DeviceA[1]`, and subsequently `DeviceA[2]` and so on and so forth. They can be configured under the same or different Communication Port, the system will take care of the rest so long they are of same name with array number.

Once a series of device is put under the same array naming, clicking on any one of them will lead to the same tag entry. Only one time of tag entry is needed and IntegraXor will do the tag data reconcilation.



Selecting on Device[2] will show entry for Device[1], indicating they are redundant.

Note

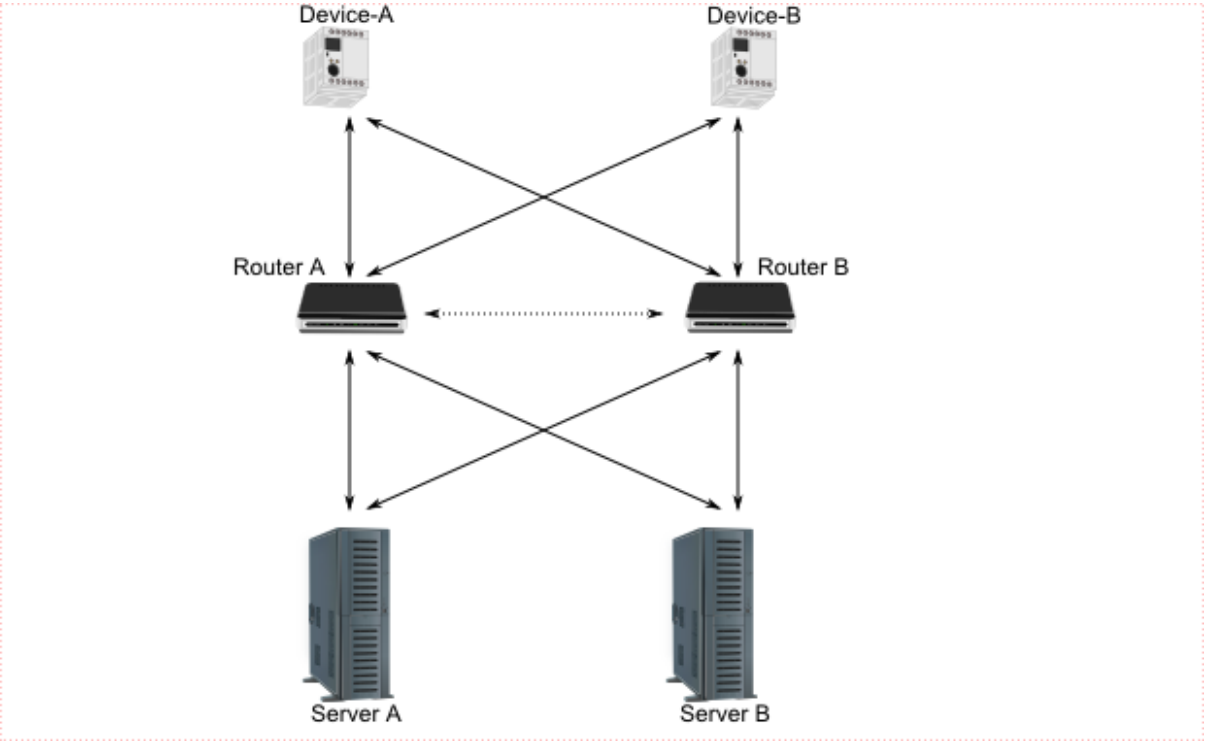
Although the system allows the device array number to be skipped, user should avoid to do so as the I/O tag count for licensing will be based on the biggest number of array.

11.3. Multiple System Redundancy

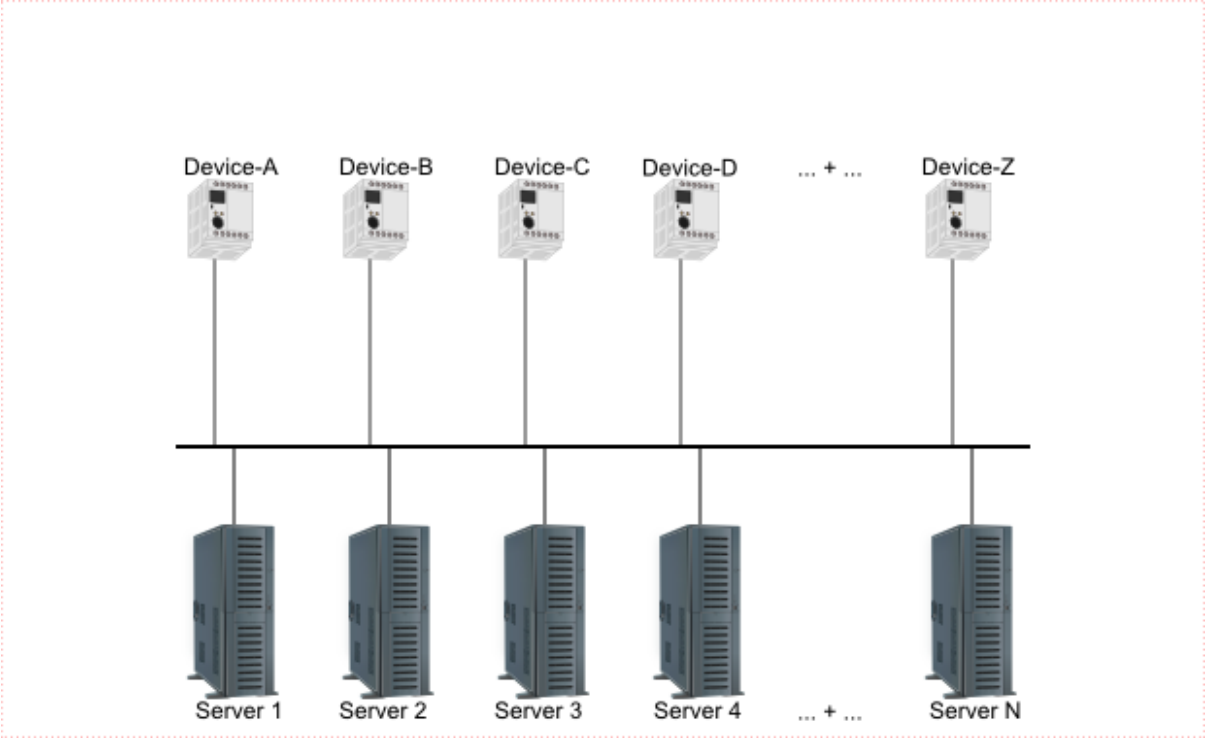
IntergraXor provides redundancy to improve the server availability, especially for mission critical operations. The server level redundancy ensures that the data is duplicated. It increases the availability of the data to the front-end view. If the primary server fails, the standby server will take over the primary role. The server with the highest ranking (lowest numerical value) among all the active peer servers in the same cluster will be promoted as the primary server. Server level redundancy covers system wide database synchronization, tag synchronization, I/O task, script task, and output task.

Database Synchronization	The server data in the database is synchronized to all peer server databases. The server data includes alarms, audits, trends, reports, persistance data, user data, and user log. The primary server is responsible to log the data into the database. The data is then synchronized to all database of the standby servers.

Tag Synchronization	Tag data is synchronized across all the peer servers. When the tag data changes, the primary server distributes the tag changes to all the standby servers. Tag synchronization includes all the I/O tags and virtual tags.
I/O, Script and Output tasks	When the server redundancy is active, the primary server runs the I/O, script and output tasks. If the primary server is down, the standby server with highest ranking will take over the primary role and run the tasks. It is based on hot standby concept.



A network configuration that shown communication path for dual Network redundancy



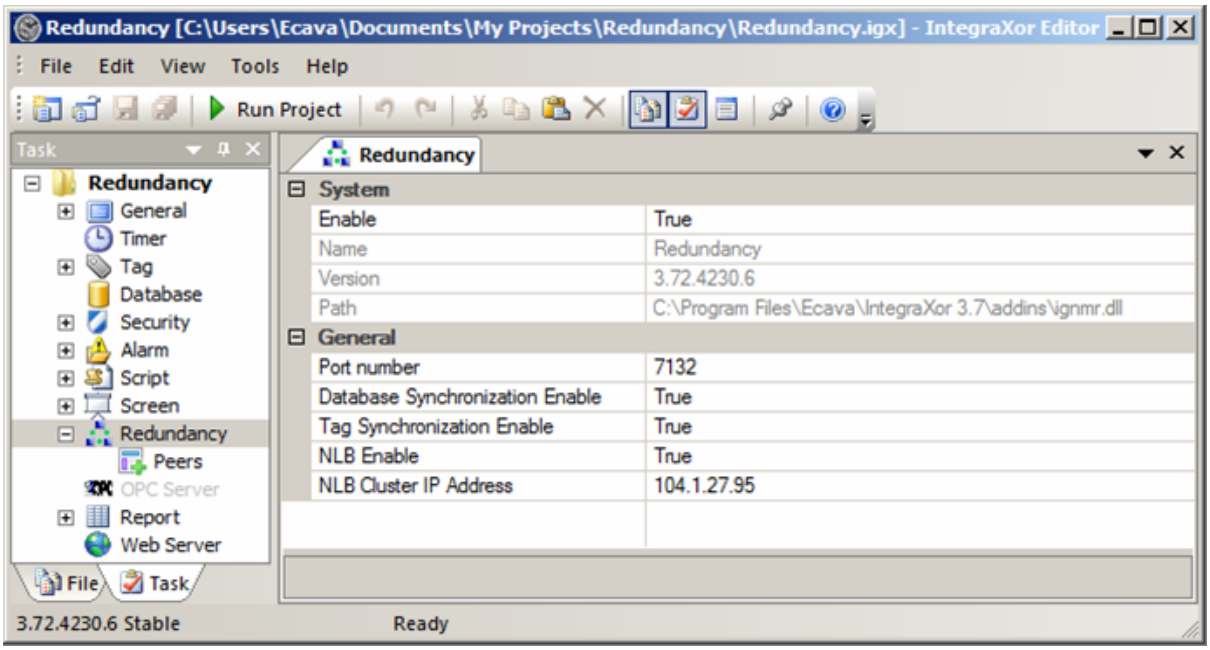
A network configuration that showing unlimited peer of redundancy

Note

Currently tested & supported database is PostgresQL. Using other brand of database will need to install its own Replication module/plugin.

11.4. System Redundancy Configuration

Parameter	Description
Port number	Network port number used for Redundancy Servers' communication channel, use default 7132 which normally won't have conflict.
Database Synchronization Enable	Enable or disable the database synchronization. Normally database has to be synchronized unless alarm, trending and report are not used, which is very very rare in SCADA project.
Tag Synchronization Enable	Enable or disable the tag synchronization. This is an option to disable the run time tags synchronization when traffic is far too congested.
NLB Enable	Enable or disable the Network Load Balancing (NLB) feature. NLB is employed to enhance the availability of the server applications.
NLB Cluster IP Address	NLB cluster IP address is a virtual IP address required when NLB is enabled. It is set identically for all the peer servers in the cluster.

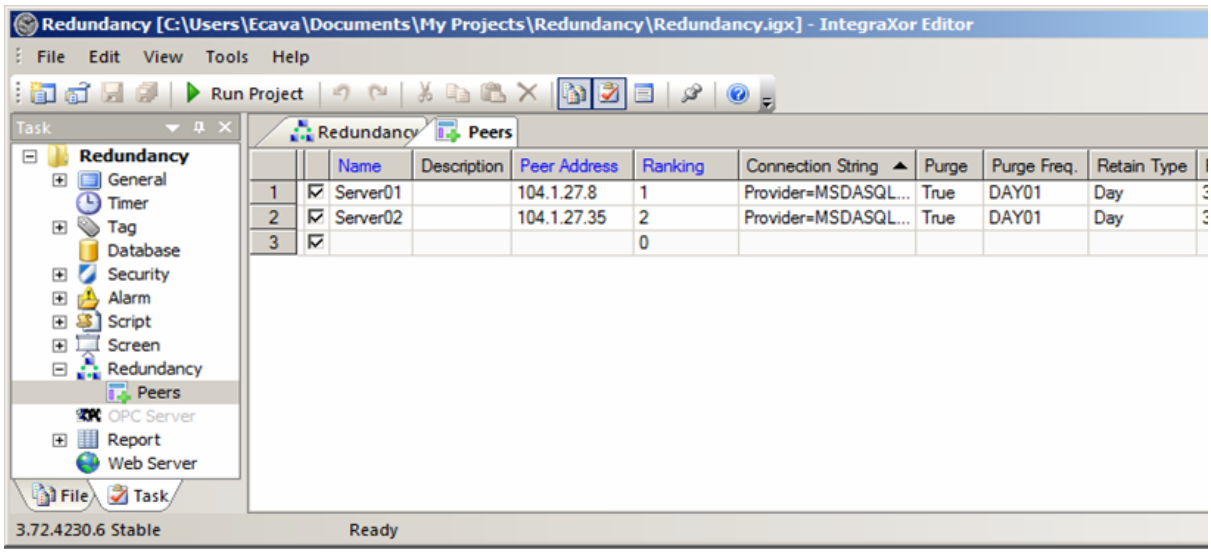


General configuring for redundancy

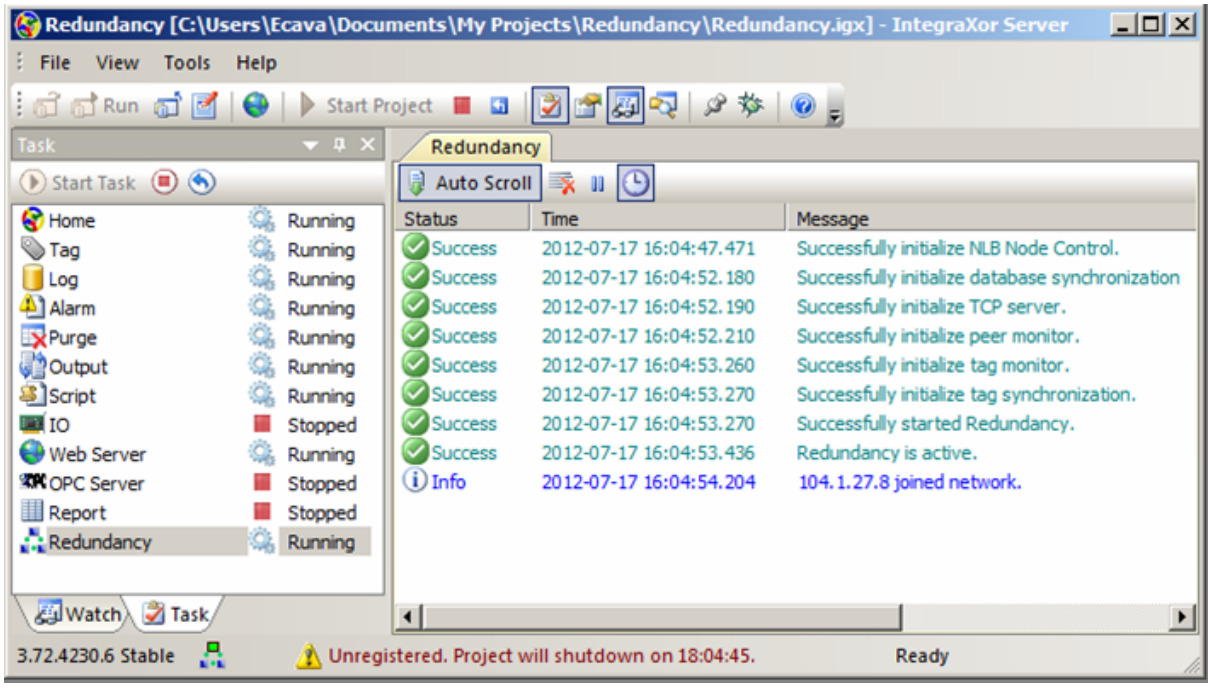
Despite the Multiple Redundancy is superior for scalability and reliability, but the configuration can be as simple as entering the IP for any peer PC that need to be participated.

Parameter	**Description**
Name	Name of the peer server.
Description	Additional information about the peer server.
Peer Address	List of IP addresses separated by comma for all peer servers.
Ranking	The Ranking parameter specifies the priority of the peer server. The peer server with the lowest numerical value among all active peers will be promoted as the primary server. The allowed values range from 1, the highest priority. This value must be unique for all the peer servers in the cluster. Only the the server being rank top as primary will be running at all time to ensure no conflit of script processing and to avoid network congestion with too much of run-time data exchange.
Connection String	ODBC connection string to connect to peer server databases. Additional database is needed for storing temporarily pending data before the synchronization is taking place.
Purge	Purge database to prevent overflow of accumulated pending data, in case the peer database is not available to be synchronized for too long.
Purge Freq.	Define how frequent old data are purged from this database. Normally setting it to wee hours like 1am to 4am shall be a good idea as it will be less busy moment for CPU.
Retain Type	Define whether old data is purged based on number of days or number of records. Data are purged in FIFO (First In First Out) basis.
Retain Count	Define how long (day) or how many entry (record) the database shall keep.

Once System Redundancy is enabled, look out the Watch List for the associated system tags that will be available for monitoring or further handling.



Configuring redundancy peer



Server is running with Redundancy task

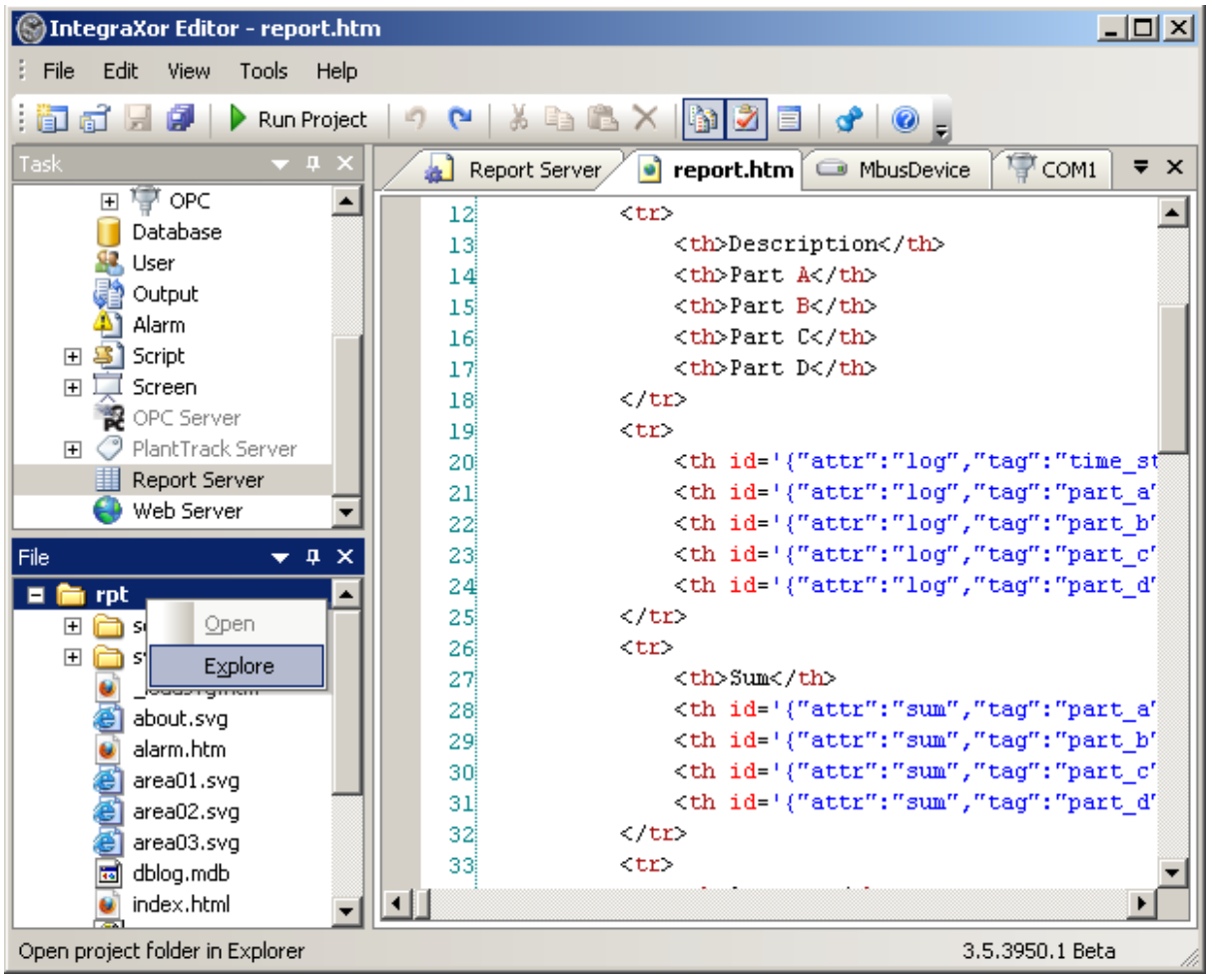
Note

Only PostgresQL [http://www.postgresql.org/] is supported for database redundancy. Use of MS SQL would need its own replication module for database synchronization.

Chapter 12. Project Management

IntegraXor has been carefully designed so its project folder is fully portable just like a web site project folder that can be placed anywhere in the file system, so long engineer avoid using fixed or absolute file path in the project development. Instead, engineer shall use relative file path to ease any future maintenance work. Backing up or restoring the project is a matter of copying the folder to the desired destination, engineer is encourange to use compressing tool for ease of maintenance although an average project folder with thousand tags won't be big as compare to many other HMI/SCADA systems. However, be aware that the default project Access database is stored within the project folder which can contribute significant file size while archiving project.

The default installation creates one main directory "ecava" at "c:\Program Files", and "My Documents\My Projects" for project creation.



Windows Explorer can be launched from within File View pane by right clicking on the desired folder.

When a web page code is updated, engineer only needs to click Refresh button or pressing F5 as shortcut. No restarting of server is needed. Since all project files are stored at server side, managing and updating of the project is centralized at server side, regardless how far the client system is. No work is needed at client side except refreshing the web page. And they are fully secured since no file is accessible at client side.

12.1. Backing up project

Backing up IntegraXor project is as simple as copying the project folder. For ease of file management,

compressing the folder into one file would be preferred. Very often compressing using 7z will yield much smaller file size, and using 7z for decompressing can be done even the filetype is not properlly named in common file formats like 7z or zip. For instance 7z can decompress file that renamed to 7z_ or zip_ for bypassing mail server.

It's pretty common that a project need to be compressed and send as email attachment. However it's not quite possible if the project contain any *.exe or *.js files as it may be blocked by many types of mail server. The best practice is to add a trailing underscore to the extension of the existing file so that the receipient would know the original file type when needed. For instance, `project.zip` shall be renamed to `project.zip_`. Likewise, the email recipient shall remove the underscore upon receiving the attachment.

Removing unused *.mdb file shall further reduce the filesize.

12.2. Migrating/upgrading project

When a new version of IntegraXor is installed, the system files in the previously created project will not be automatically updated. User shall first backup the exiting project before migration and then simply create a new project based on newly installed IntegraXor. And then copy the whole system folder from new project to existing project. It's best to replace also `index.html`, `alarm.htm` and `user.htm` from new project to existing project. In some cases, user may also need to replace `plot.htm`.

Starting from version 4, system files have been embeded into server so system folder no longer needed to be placed within every project folder. IntegraXor will automatically use the latest built-in system folder within itself. This will simplify the project upgrade process.

Starting from version 4.1, script task with ActiveX disabled will be case sensitive for internal functions call. So `setTag()` cannot be written as `settag()`, and `getTag()` cannot be written as `gettag()`.

12.3. Group development

Concurrent access from multiple user to single project is not allowed. If a large project requires concurrent development, project manager could split the data entry works to multiple engineers and use export/import to merge the works later. Alternatively, multiple engineers could do concurrent access to cloud based spreadsheet and then import into project later.

12.4. Updating Template

Project template is actually a zip file for a standard project folder. It resides in system directory where IntegraXor is installed, default directory shall be "C:\Program Files\Ecava\IntegraXor 4.1", but the version number could varies. New template will be released from time to time along with installer when new themes or skin and features are available.

In the case that engineer wishes to customize their own template, simply modify all aspects accordingly, and then compress the whole project as template.zip and replace the original `template.zip` as described. Project Editor will use this customized `template.zip` as the base for the next project creation.

12.5. Suggested Supporting Tools

Just like any web site project, IntegraXor project mainly consist of text files for HTML pages, styling and scripting. Other than configuration file and raster image files, engineer shall be able to manage those text files with many useful external tools that freely availabe. For instance, engineer could compare two project folders with WinMerge [http://winmerge.org] for differences, engineer could also use Agent Ransack

[http://www.mythicsoft.com/agentransack] for text content searching in Windows XP environment.

Below are a list of tools from external developer. Usage of these tools are solely on engineer responsibility and no support is provided from IntegraXor, although we may help wherever possible. Please check out its terms and conditions, licence agreement. We also encourage user to donate and support to these developers should the softwares are found useful.

Purpose	Tools and Description
Text Editing	There are many free and excellent text editor like "ConText" which can be downloaded at http://www.context.cx or "Notepad++" at http://notepad-plus.sourceforge.net which also has another portable version that can be found at http://www.portableapps.com/
Revision Control	A serious and careful engineer may like to use revision/version control for their work for traceability. "SVN" can be found at http://subversion.tigris.org.
Revision Control GUI	Engineer who use Subversion may need Tortoise for supplement. http://tortoisesvn.tigris.org
Comparison	If a full blown version control system is found overwhelmed by an engineer, a simple text comparison is recommended for simple file comparing and merging. "WinMerge" can be downloaded at http://winmerge.org.
Search	In Windows XP, the text content searching for unknown file is not possible in default system setting. Engineer may need a content searching tool like "Agent Ransack" which can be downloaded at http://www.mythicsoft.com/agentransack
Screen Positioning	"This Screen Ruler is a great tool that allows you to accurately measure anything on your screen. This is especially useful when measuring graphics, web page browser sizes or whatever." http://www.spadixbd.com/freetools/jruler.htm
Color Picker	"Pixie" is a color picker that could tell the color in hex, RGB, HTML, CMYK and HSV values of a pointed object on screen. http://www.nattyware.com/pixie.html

Warning

There's no way that the supports and responsibilities of the above mentioned development tools/products could be associated to IntegraXor Server.

12.6. Working with Database

By default, Microsoft Access flat file database is included in a new project when it's created using Project Editor. A database is a basic requirement in a SCADA application. Typically used to store data for Trending, Alarms, Tag Persistence or Report Archiving.

If a more reliable and larger database than Microsoft Access is required, IntegraXor can log to any ODBC-compliant database such as Microsoft SQL, Oracle, MySQL and PostgreSQL. Microsoft has released SQL Express for free with 10 GB data storage limitation, which is sufficient in most applications. It has toolset for database management which can be used to create, modify and view tables. Use this Management Tool to create customized tables for additional database logging requirement or to view logged data. It can be downloaded at http://www.microsoft.com/express/Database/.

To integrate with external databases, engineer need to add an entry into database task, with its relevant "Connection String". A comprehensive list of Connection Strings for various database can be found on this website: http://www.connectionstrings.com/. Commonly used connection string is listed in the following table.

Database	Connection String

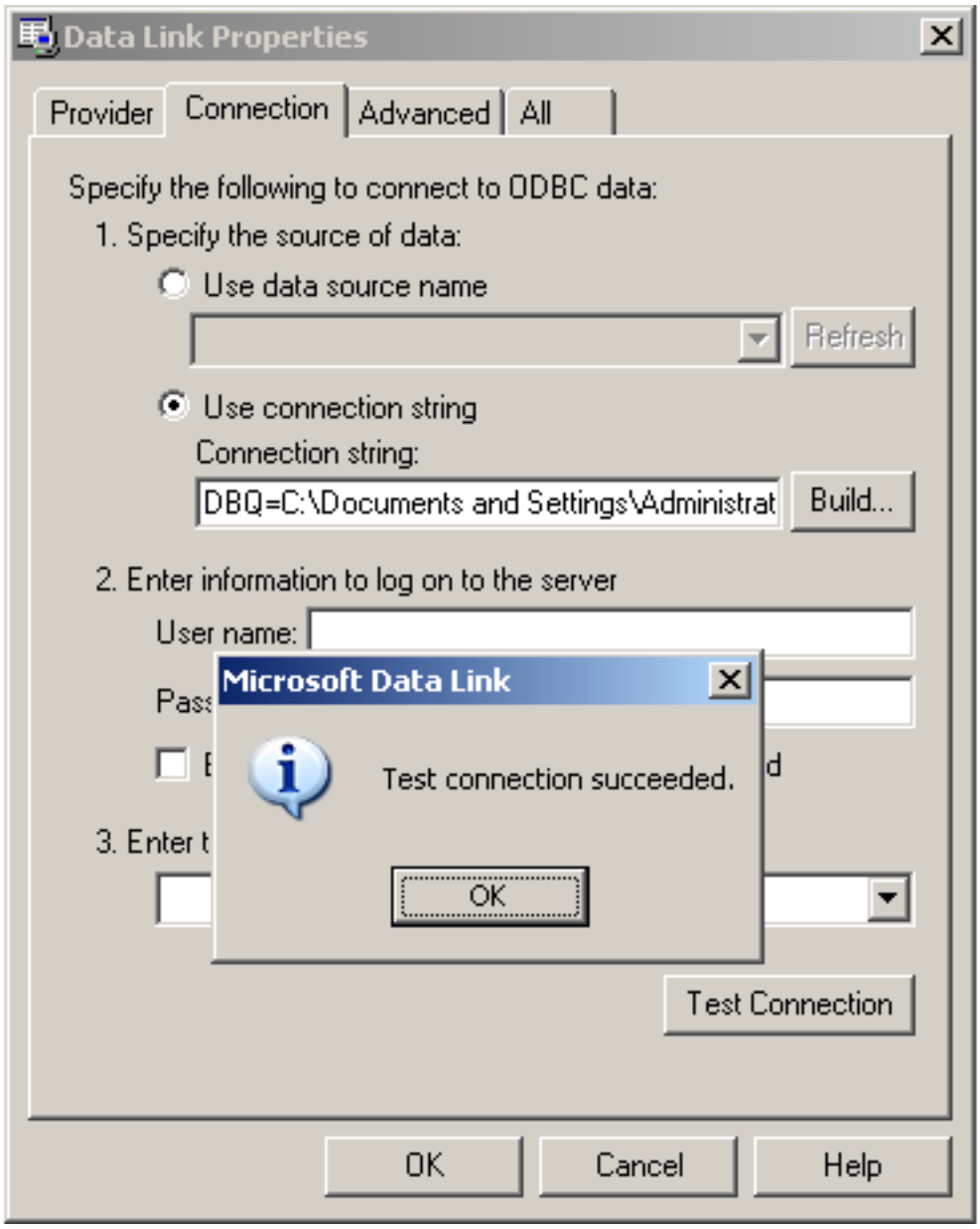
Microsoft SQL Express	`Provider=MSDASQL.1;`
	`Extended Properties="DRIVER=SQL Native Client;`
	`SERVER=<COMPUTER_NAME>SQLEXPRESS;`
	`DATABASE=my_job;`
	`trusted_connection=yes";`

PostgreSQL	`Driver={PostgreSQL};`
	`Server=IPaddress;`
	`Port=5432;`
	`Database=myDataBase;`
	`Uid=myUsername;`
	`Pwd=myPassword;`

MySQL	`Driver={MySQL ODBC 5.1 Driver};`
	`Server=localhost;`
	`Database=myDataBase;`
	`User=myUsername;`
	`Password=myPassword;`
	`Option=3;`
	
	Note
	Be aware the ODBC Driver version could be 3.51, or other numbers.

Important

Be aware the Connection Strings are shown in multiple line for better readability but must be entered as single line in actual use.



Clicking browse button [...] shall launch the Data Link Properties which could Test Connection on the string entered.

After the database is added, tables need to be created by executing its respective SQL script which can be obtained accordingly for MS SQL [http://www.integraxor.com/developer/db/mssql.sql], MySQL [http://www.integraxor.com/developer/db/mysql.sql], PostgreSQL [http://www.integraxor.com/developer/db/postgres.sql] and Oracle [http://www.integraxor.com/developer/db/oracle.sql].

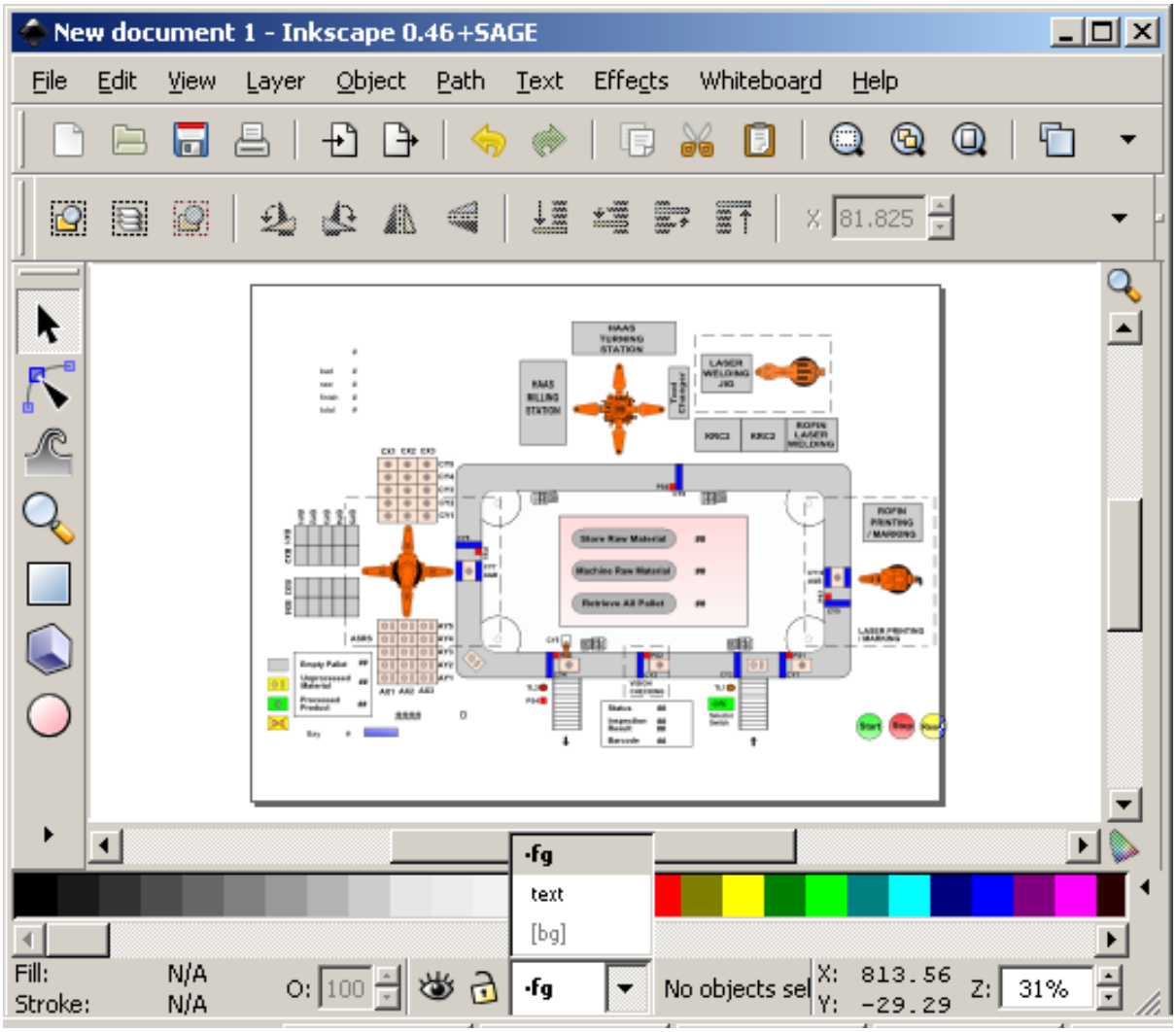
12.7. Working with Inkscape

Clicking on Help - Inkscape Manual shall lead to the main reference of this designer grade drawing tool. Although the icons are well placed and obvious with its function, but memorizing the shortcut keys for commonly used tools or operations is really the key to fast graphic editing. The shortcut key listing is available at http://inkscape.org/doc/keys046.html, which is also accessible via menu of Help - Keys and Mouse Reference.

Many good SVG examples and librabries can be obtained from http://openclipart.org. There are already too many good documentations online for inkscape. One commonly needed graphic object is a button which can be found in this step by step online tutorial http://tavmjong.free.fr/INKSCAPE/MANUAL/html/Button.html.

12.7.1. Multiple Layer

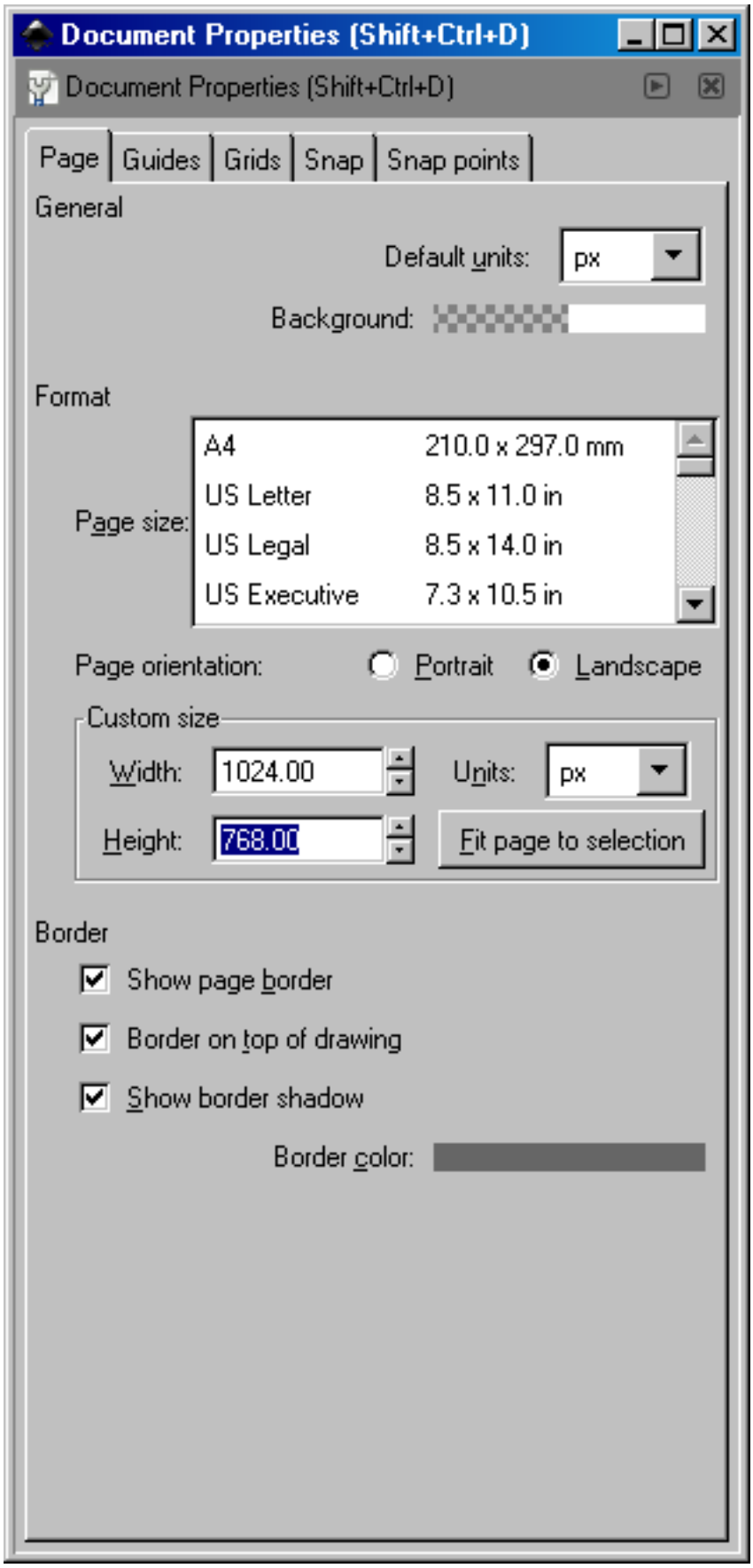
It is always good to create at least one additional layer for use as background purpose. It can contain just one huge rectangular or with some background design with it. It should be separated from the foreground work so that it can be locked within that layer and can be ignored when working on the foreground drawing. Engineer shall consider to have even one more layer for text only if the drawing is complicated, so that text can be animated to be hidden in overview and show only upon zoom in.



Layer Control is at Mid of the Bottom Panel

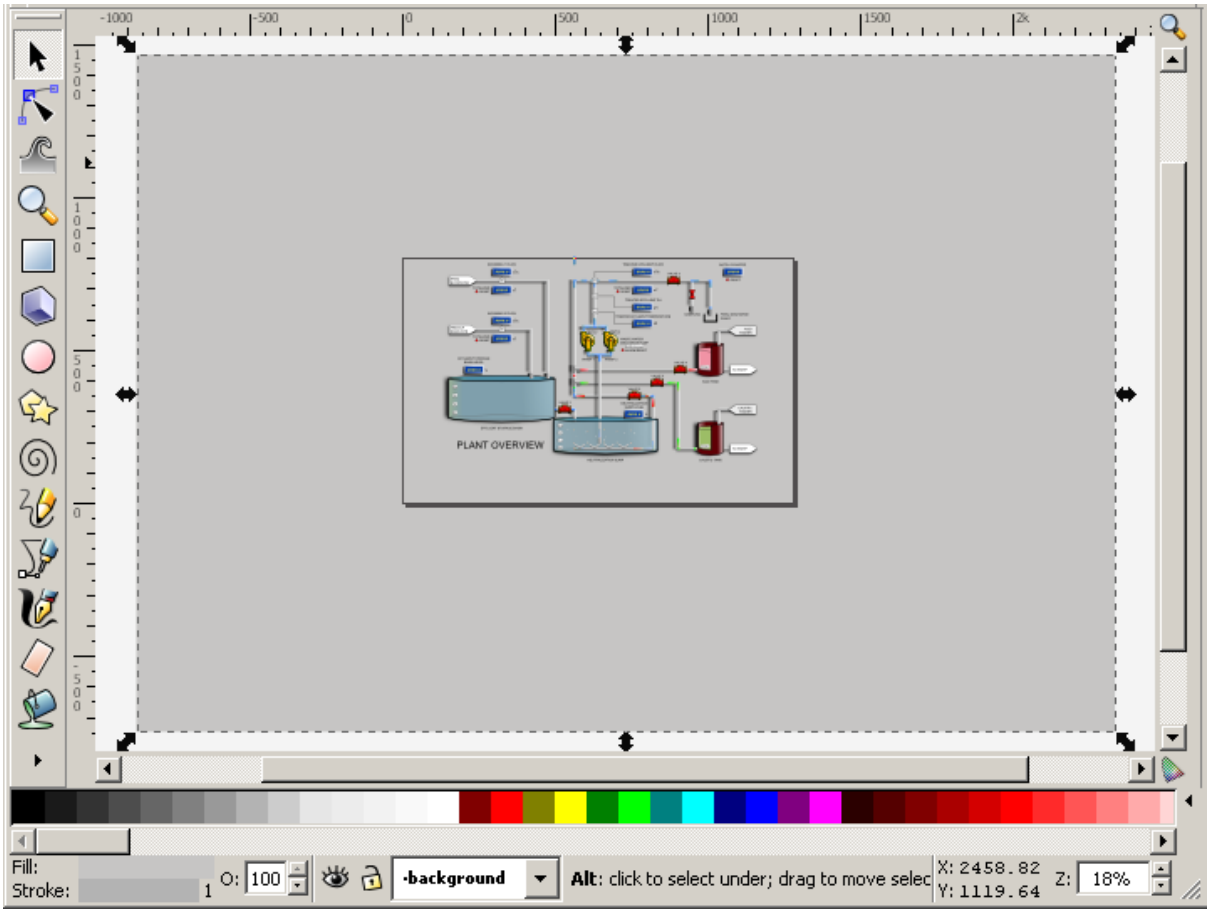
12.7.2. Resize and Background

Note that IntegrXor has an auto-fit feature which will always ensure the whole mimic could be seen in any size of window, monitor or resolution. As such engineer don't have to worry about the finally used monitor size or screen resolution, and don't have to bother one same screen to be view over different sizes of monitor across different places.



Document properties.

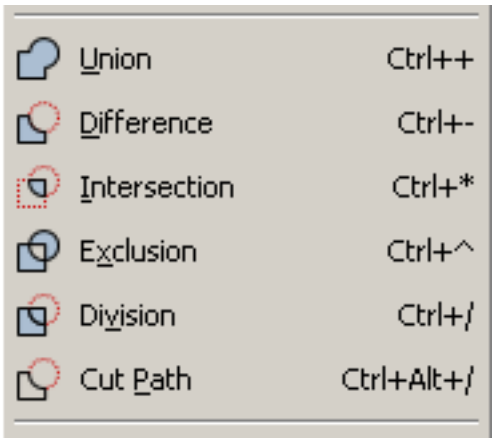
Engineer is then advised to turn on the visibility of drawing border like the screen shot and always draw within the border only. However, if a rectangular background is needed then it should be drawn beyond the border with excessive size, so that no edge of white spaces will be shown when screen is resized. And since space outside of border might be shown when window or screen size is not matched with the mimic, engineer shall avoid to place any temporarily item or clone's origin outside of the border. Instead should consider creating one extra layer to keep temporary item to which shall set to hidden.



Display will always fit to border, so background rectangle is best to drawn three times the height and width.

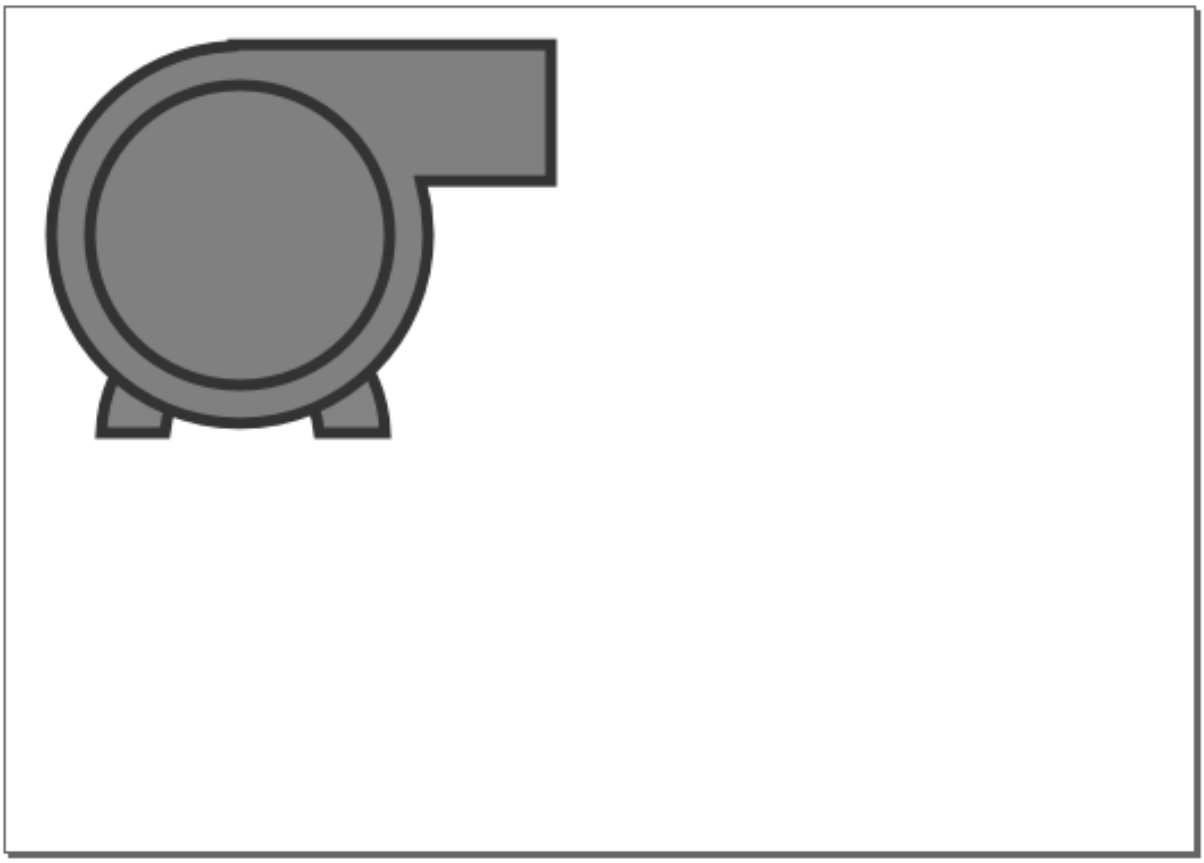
12.7.3. Complicated Shape

Two most useful shape drawing tool in Inkscape is Rectangle and Circle. What if a single peace of more complicated shape is needed? First tool that comes into mind may be using drawing pen for Bezier Curves & Striaght Lines, however, using Path Union Operation & Path Difference Operation would be easier option. These operations can be found under Path menu, but it's pretty convenient to associate the operations to the shortcut key of Ctrl + Plus & Ctrl + Minus respectively. In most cases, engineer only needs to use these two "combining" and "cutting" actions to achieve the desired shape, so although the other path operations are more direct to the result but might be rarely used. But then again this is very depends on personal preference.

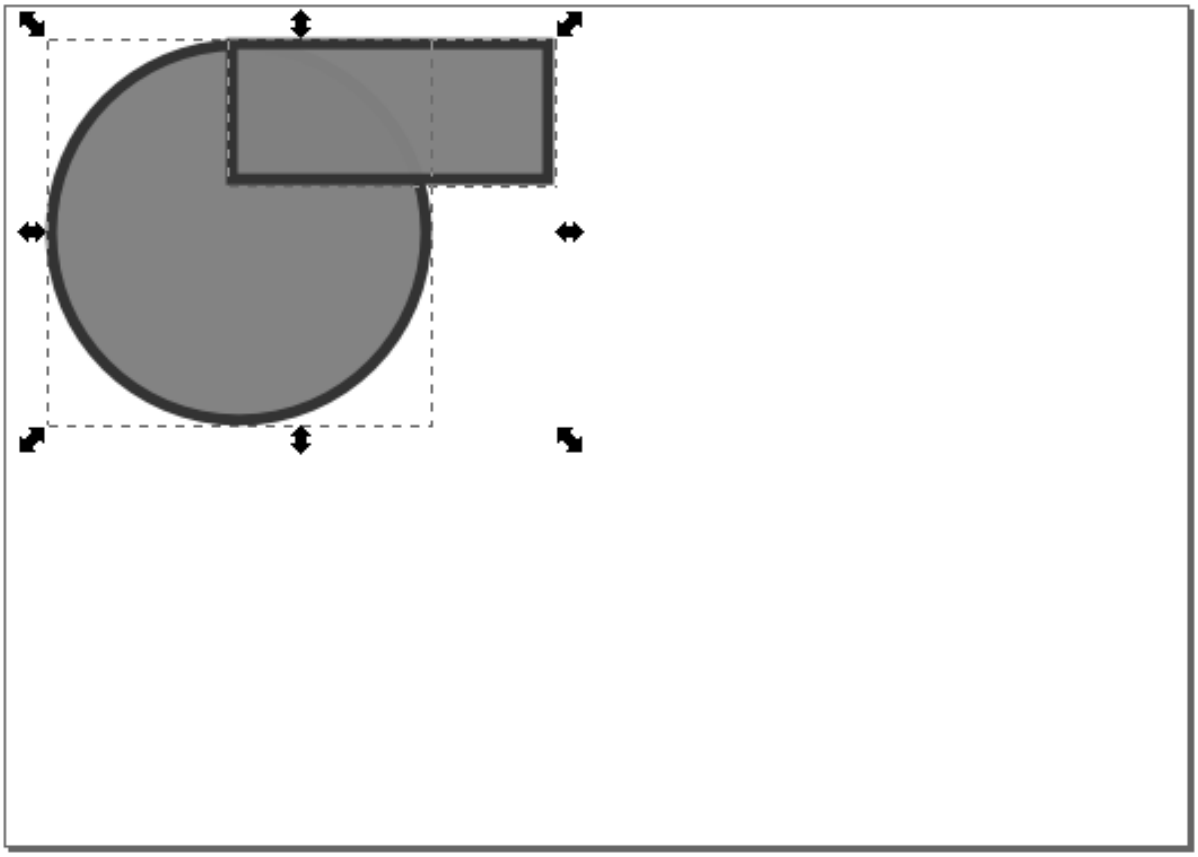


Path Operations Menu.

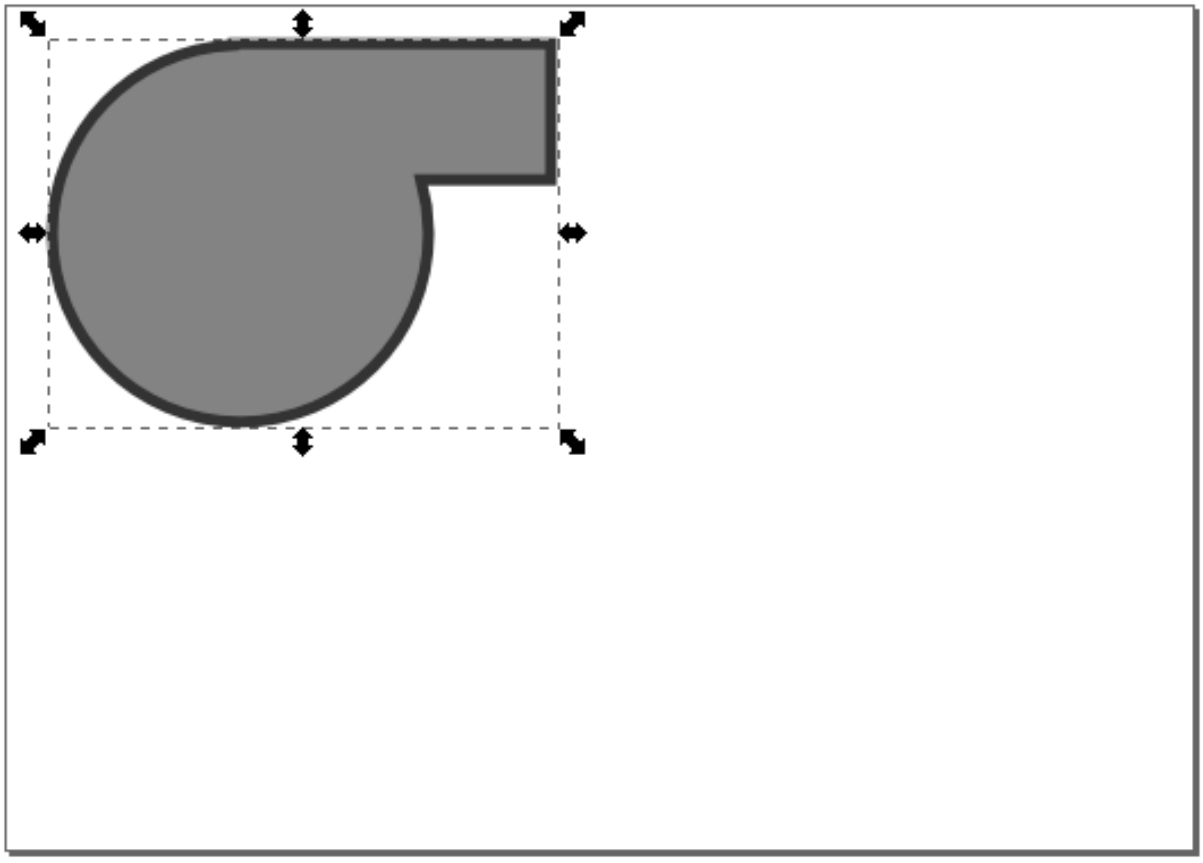
Below is several demonstration of Path Operation steps in drawing a simple pump symbol like below.



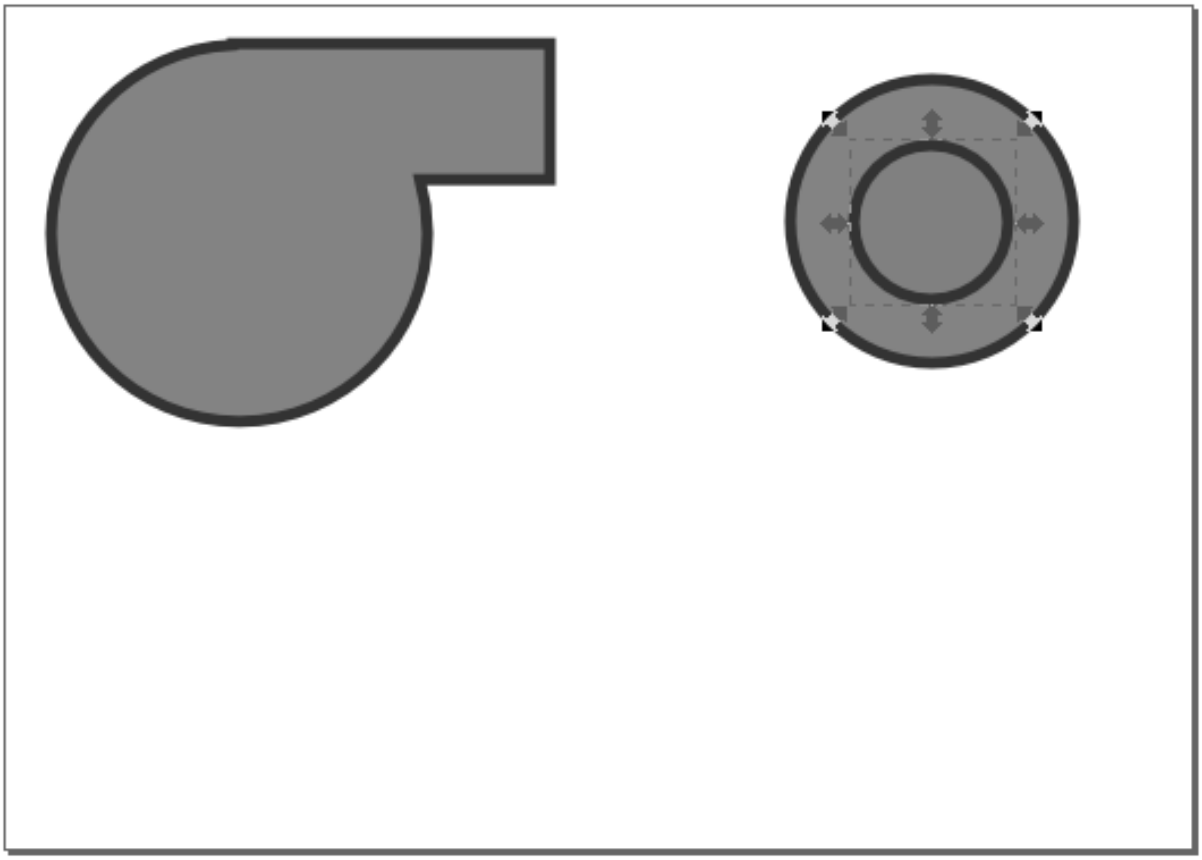
A simple pump symbol to be drawn.



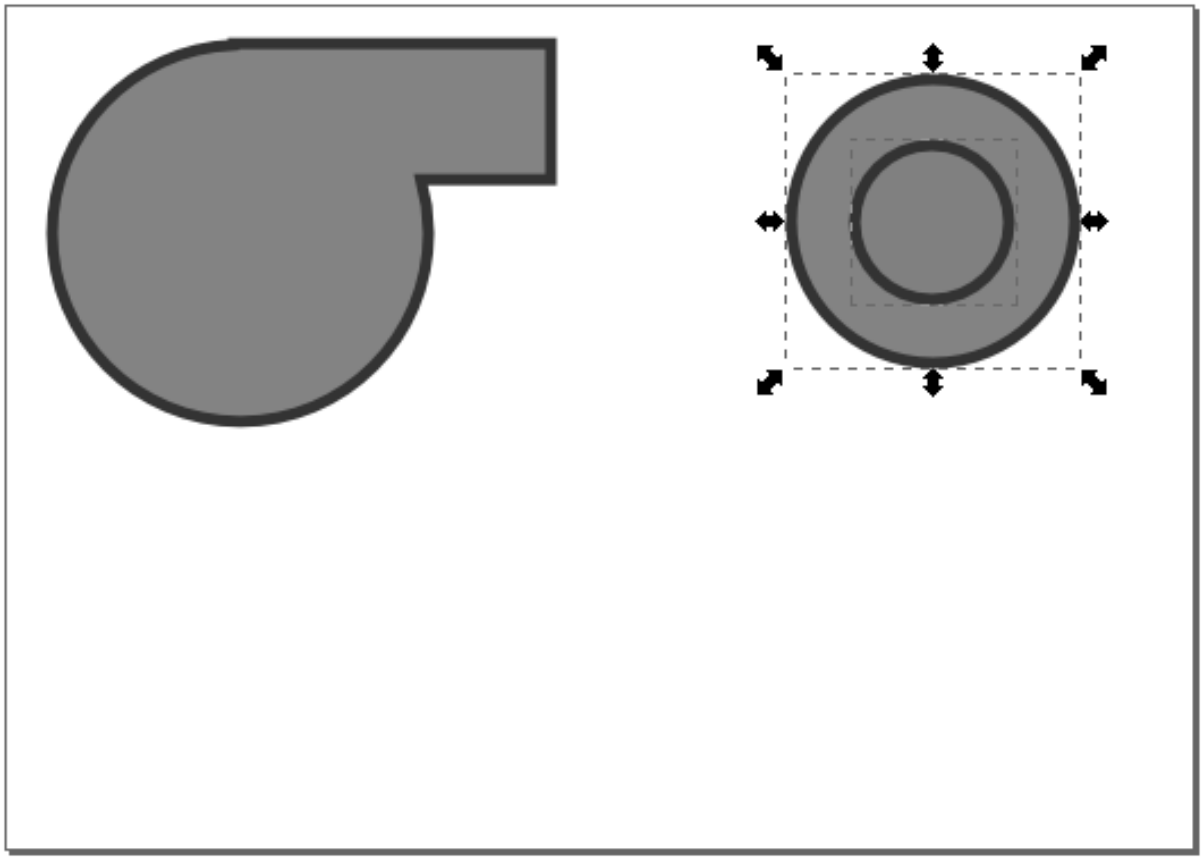
Step 1. Draw the desired shapes for pump body and highlight them.



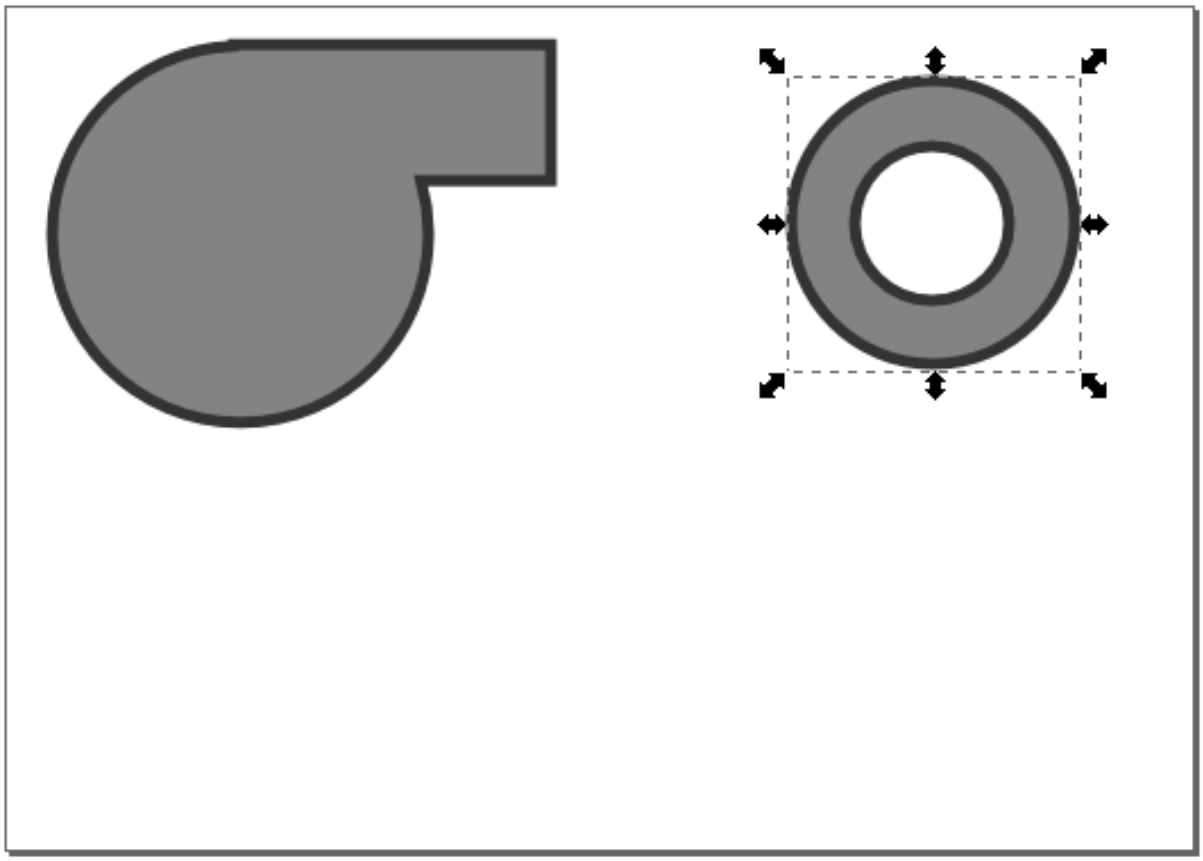
Step 2. Hold down Ctrl key then press '+' to perform Path Union operation.



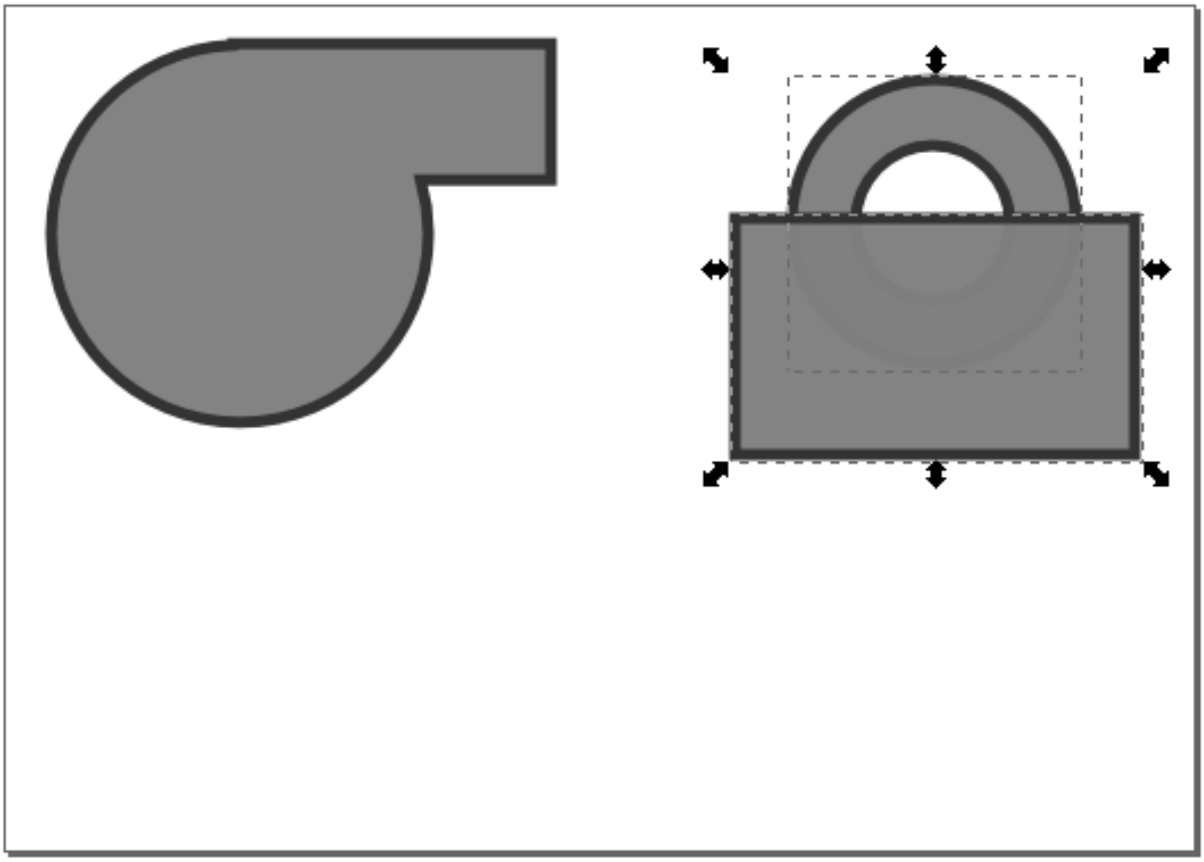
Step 3. Draw the legs by first drawing one big circle then place another smaller circle on top.



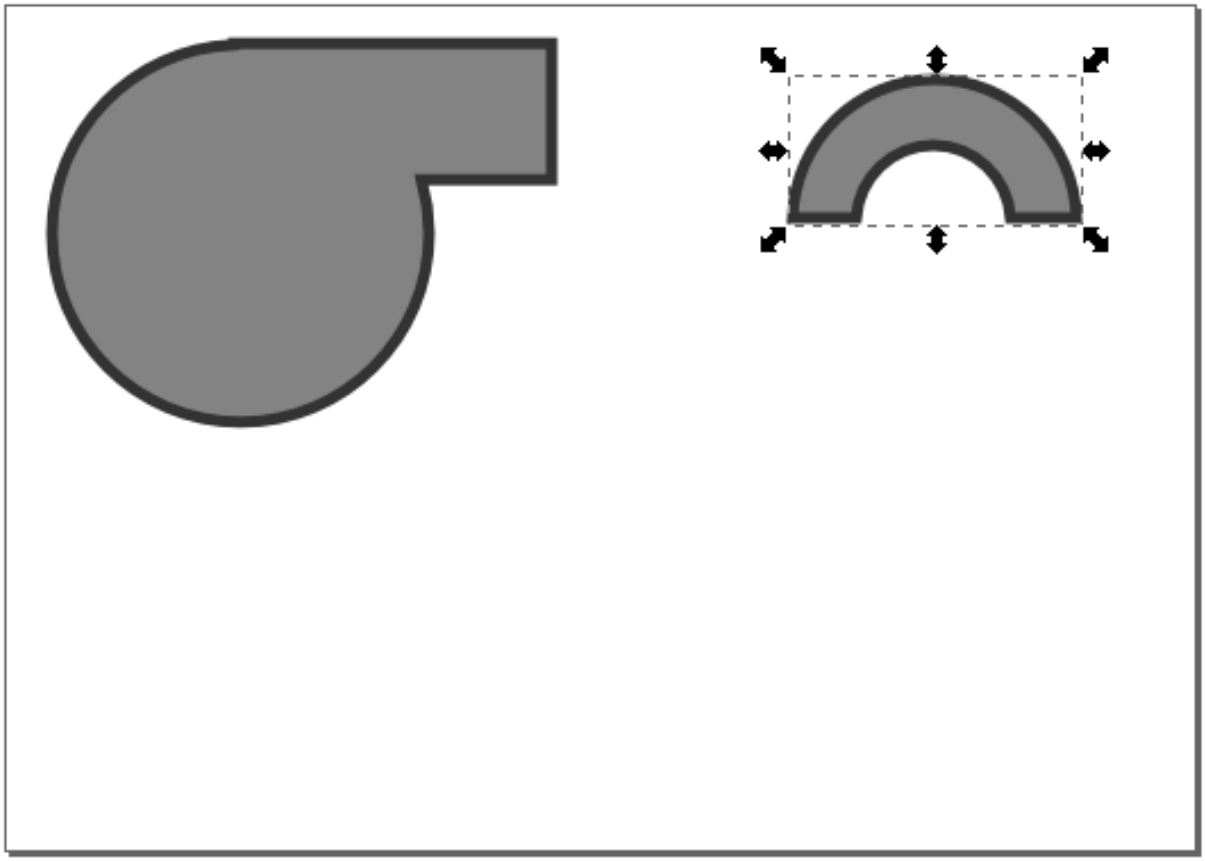
Step 4. Highlight two circles then hold down Ctrl key and press minus (-) to perform Path Difference operation.



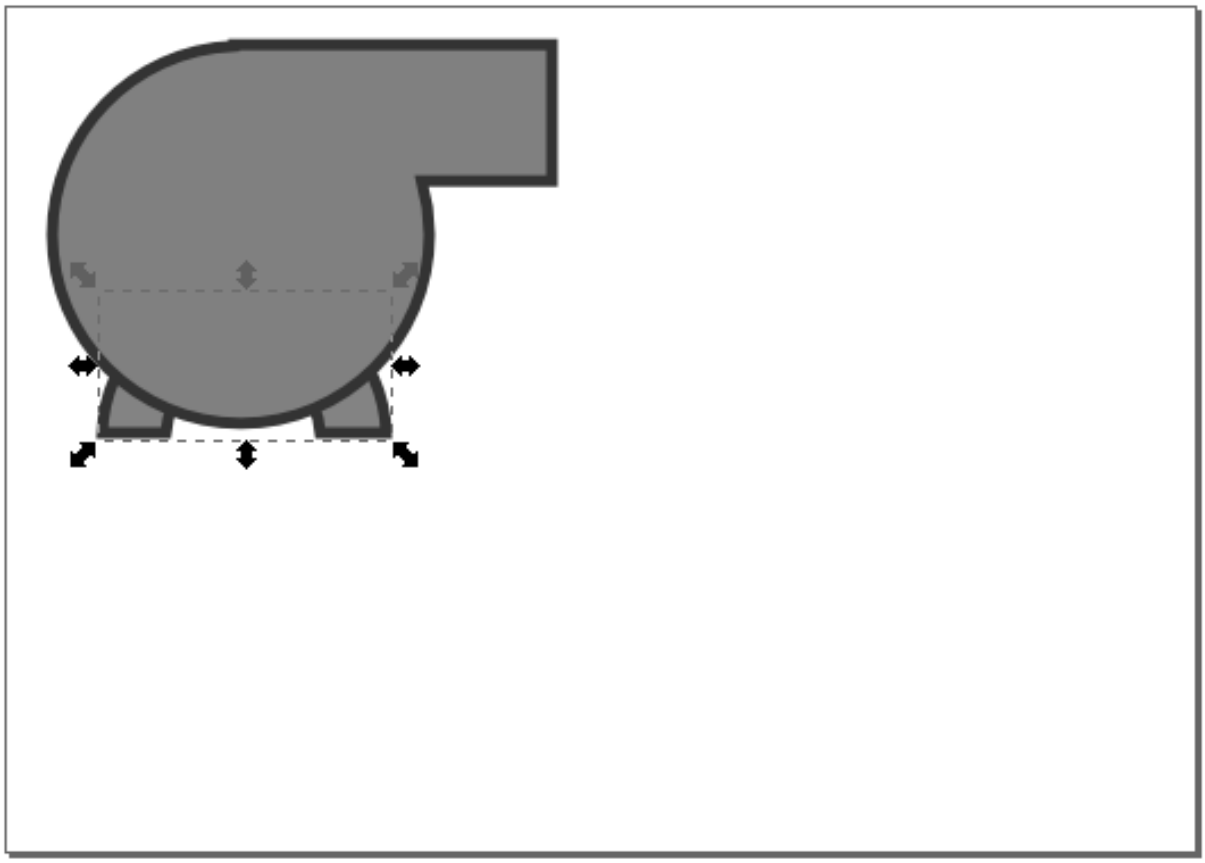
Step 5. The result is the big circle has a cut out of small circle.



Step 6. The next step is to draw another rectangle and place it on top then highlight both of them.



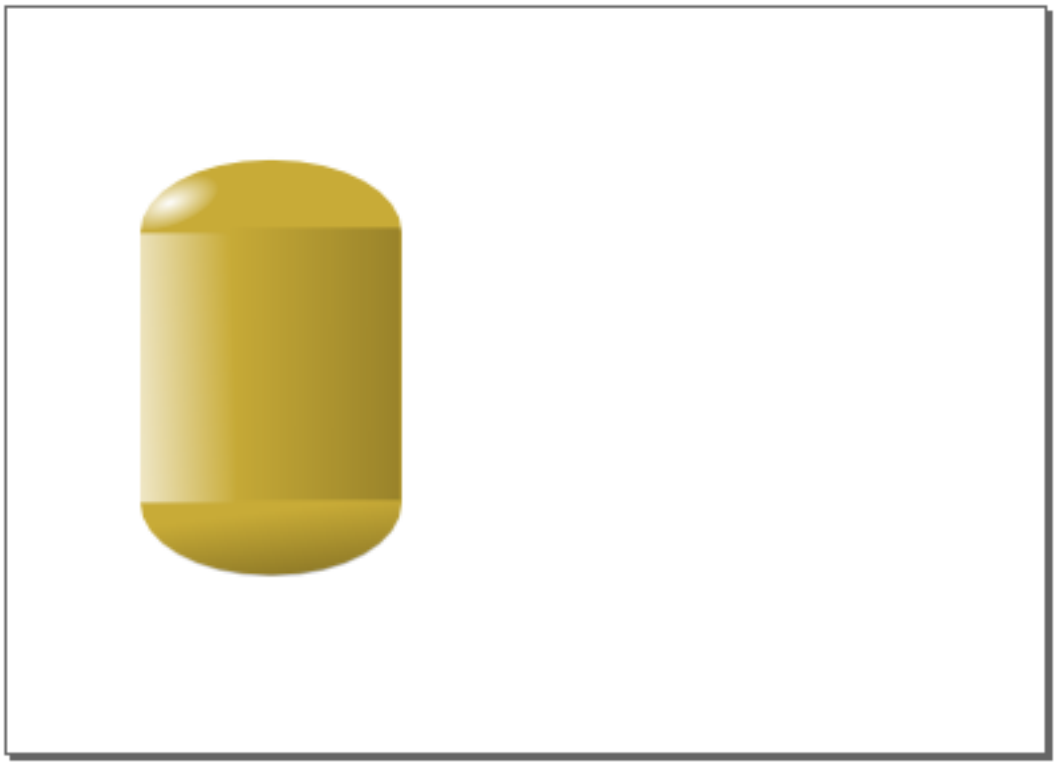
Step 7. Doing another Path Difference operation will cut off the bottom part.



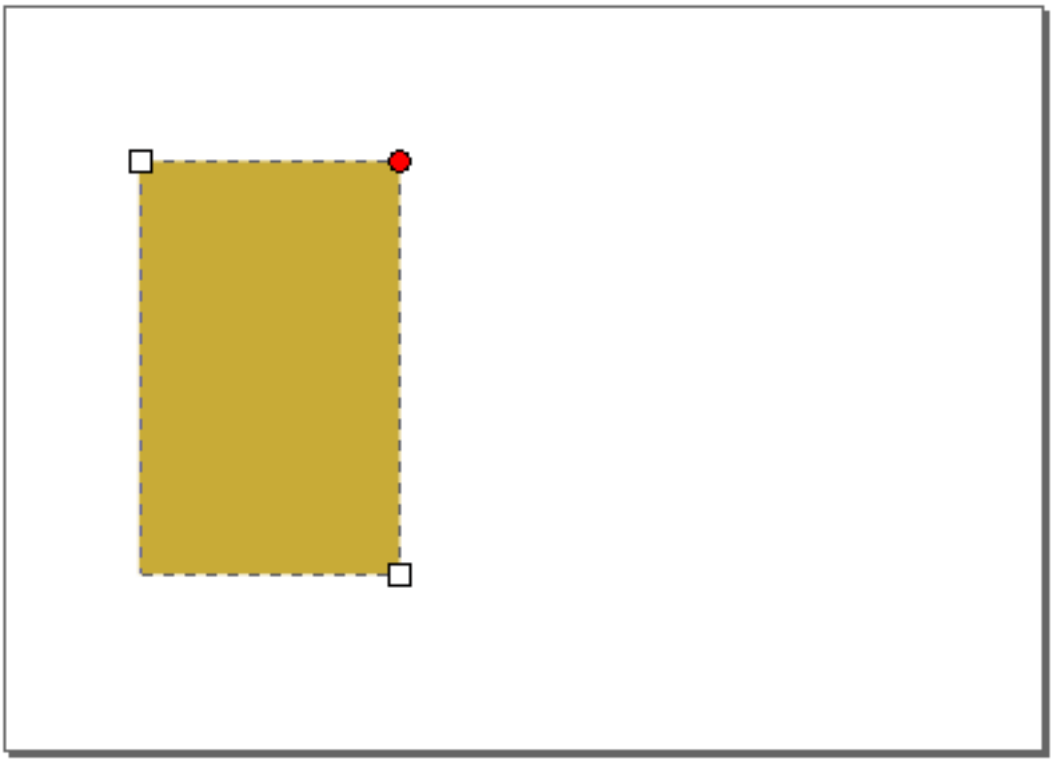
Step 8. Place the leg behind the pump body, then draw another circle to place on the pump body shall complete this simple demo.

12.7.4. Gradient Effect

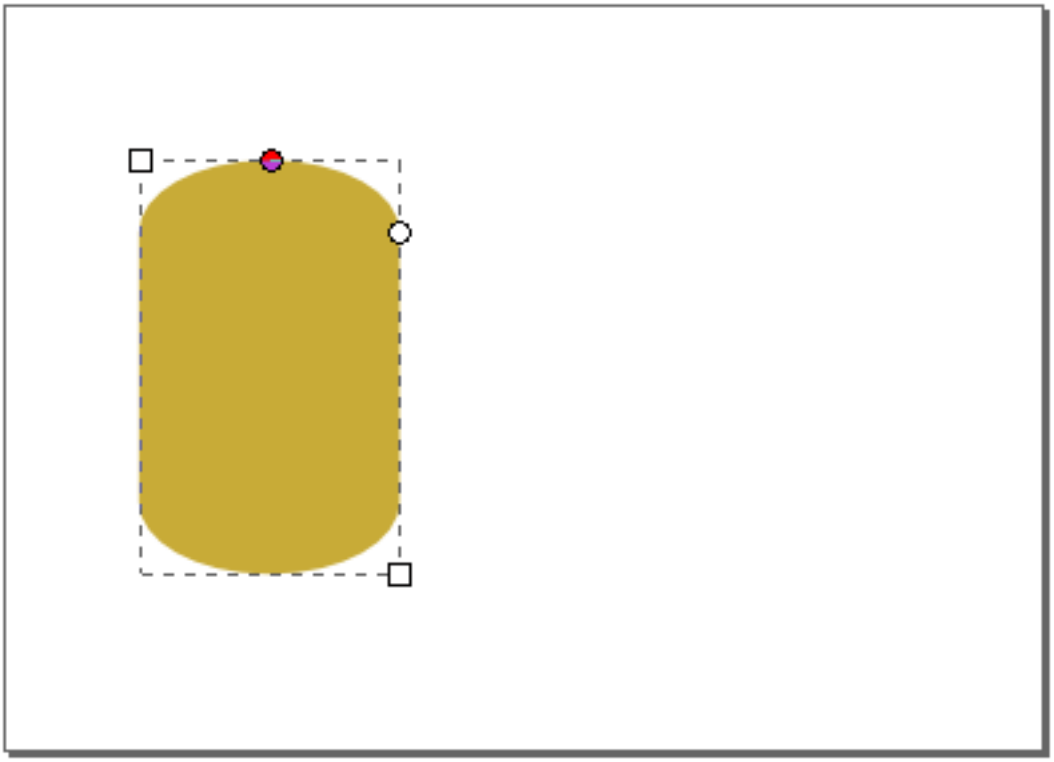
Gradient effect gives a very special impression to an ordinary graphic, a correct touch up could make an object become lively and outstanding. It produces very sharp and nice resolution as compare to traditional SCADA that plays trick on uses several peaces of object with different color shades. This traditional trick is difficult to produce yet the resolution is very bad. As such traditional SCADA system normally shipped with huge graphic library pack. Below are several quick and simple steps that demonstrate how to draw a tank with impressive look.



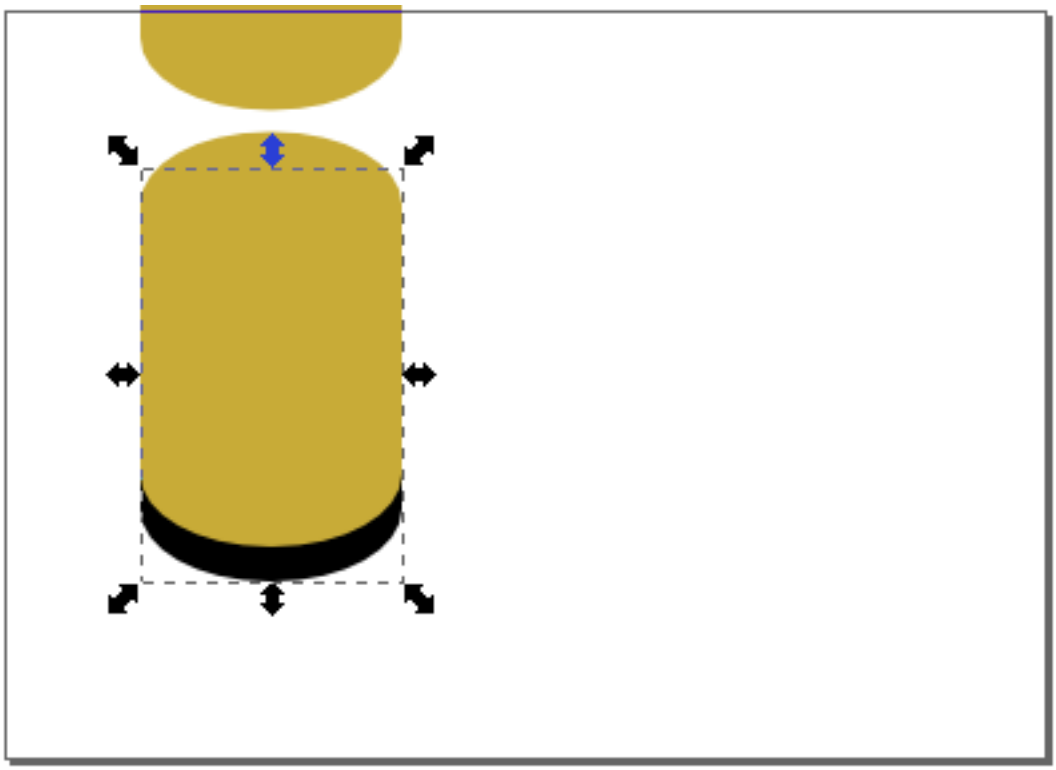
A simple tank with gradient effect.



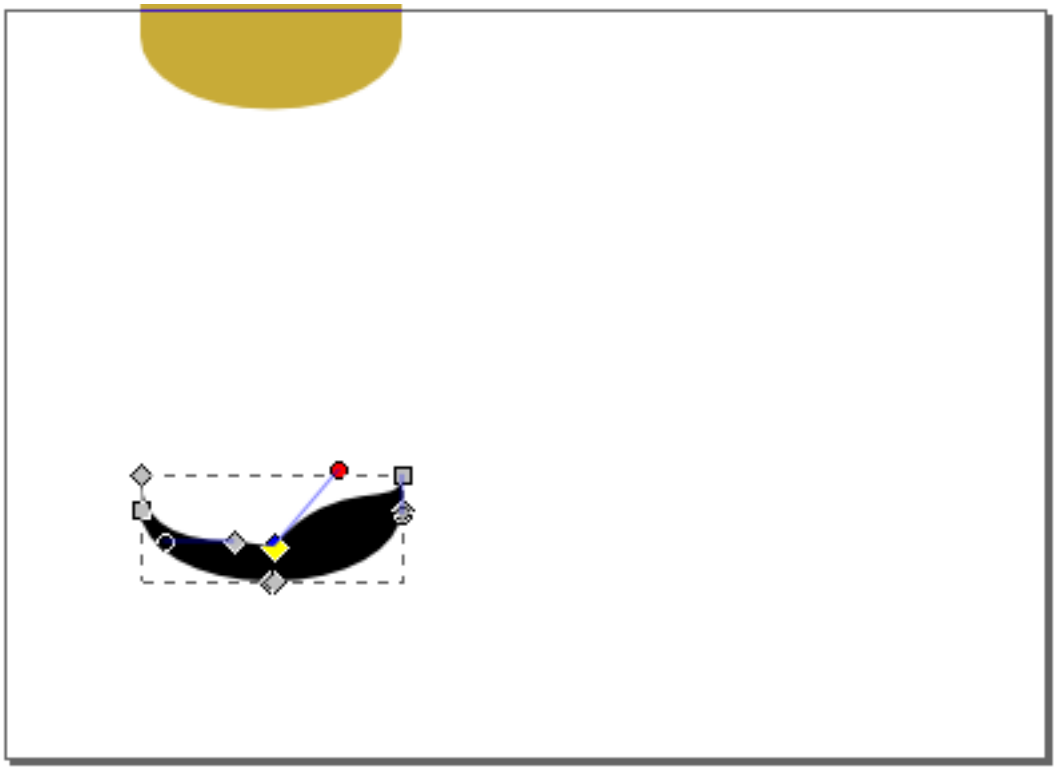
Step 1. Draw a rectangle without stroke then double click to get into resizing mode.



Step 2. Simply drag on two small circles to shape the tank.

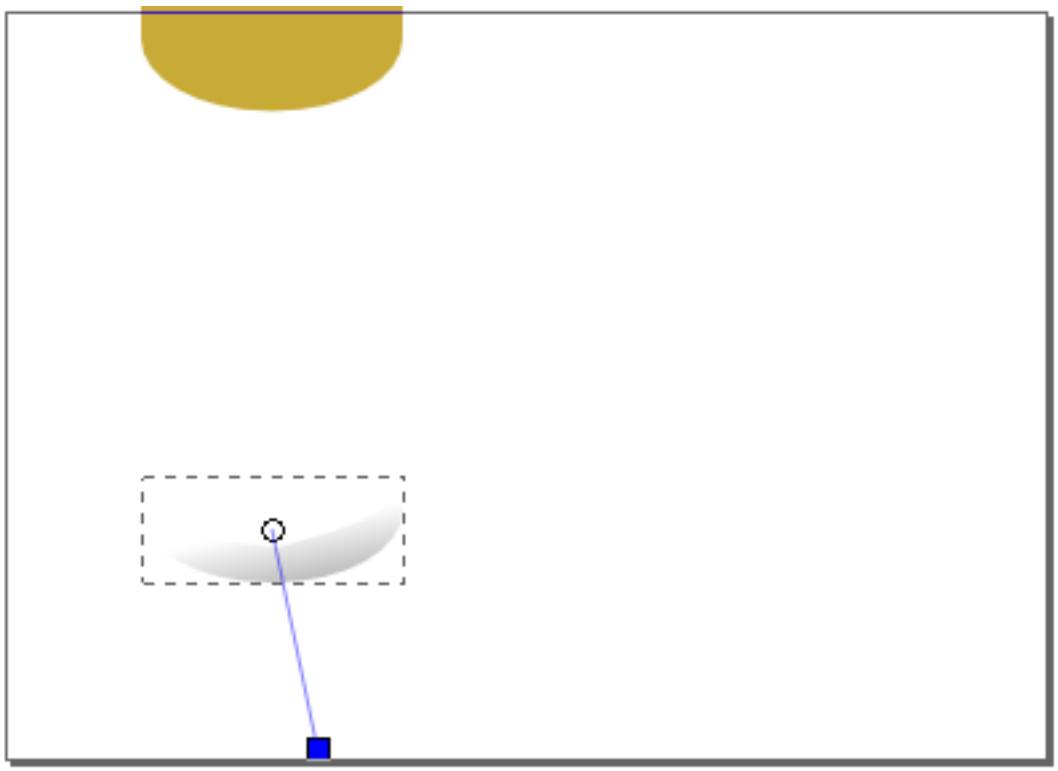


Step 3. Press Ctrl + D for twice to duplicate two more similar shape of objects. Temporarily move one aside to the top of the diagram. Then move another one slightly above the existing object. Set the existing object to black. Holding down Ctrl while dragging could ensure object always move along the same line from the original position.

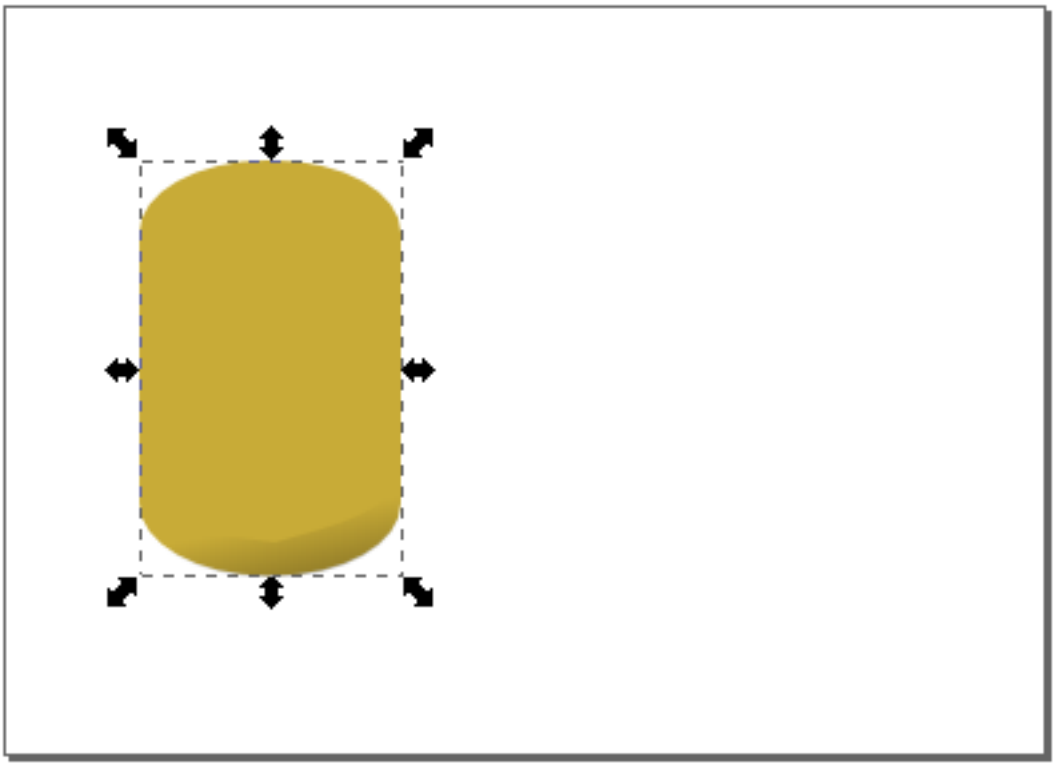


Step 4. Highlight both of them and Press Ctrl + Minus to perform Path Difference Operation. Double clicking

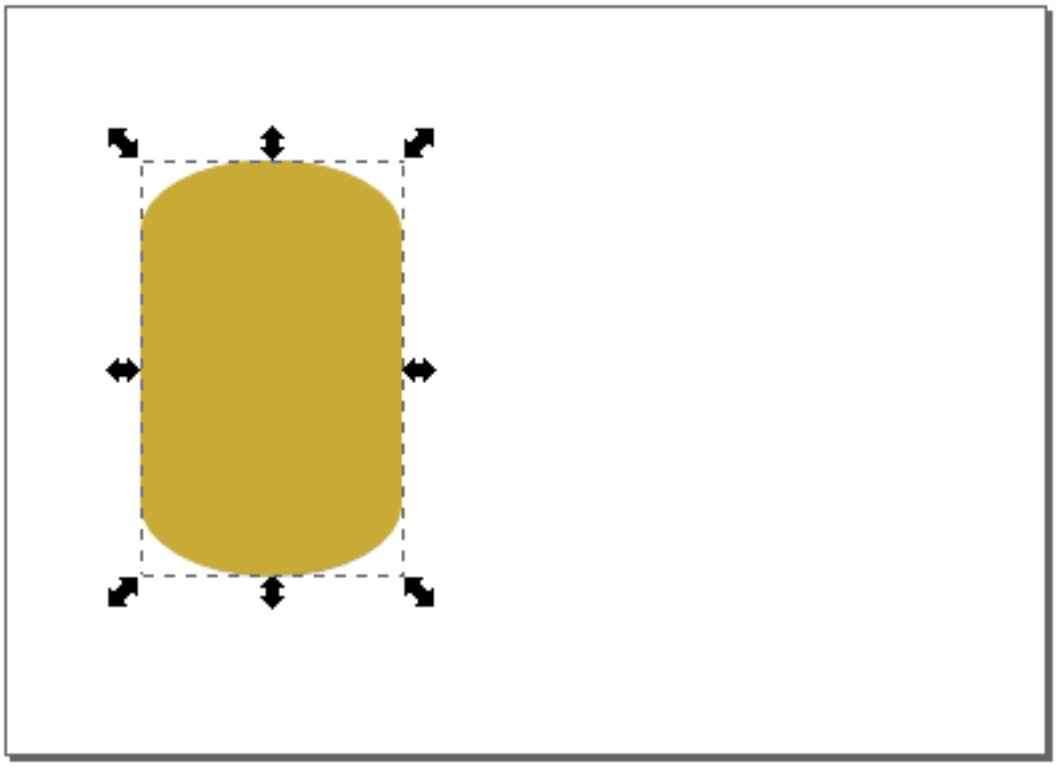
on the result to further fine tune the object to the desired shape, only if required.



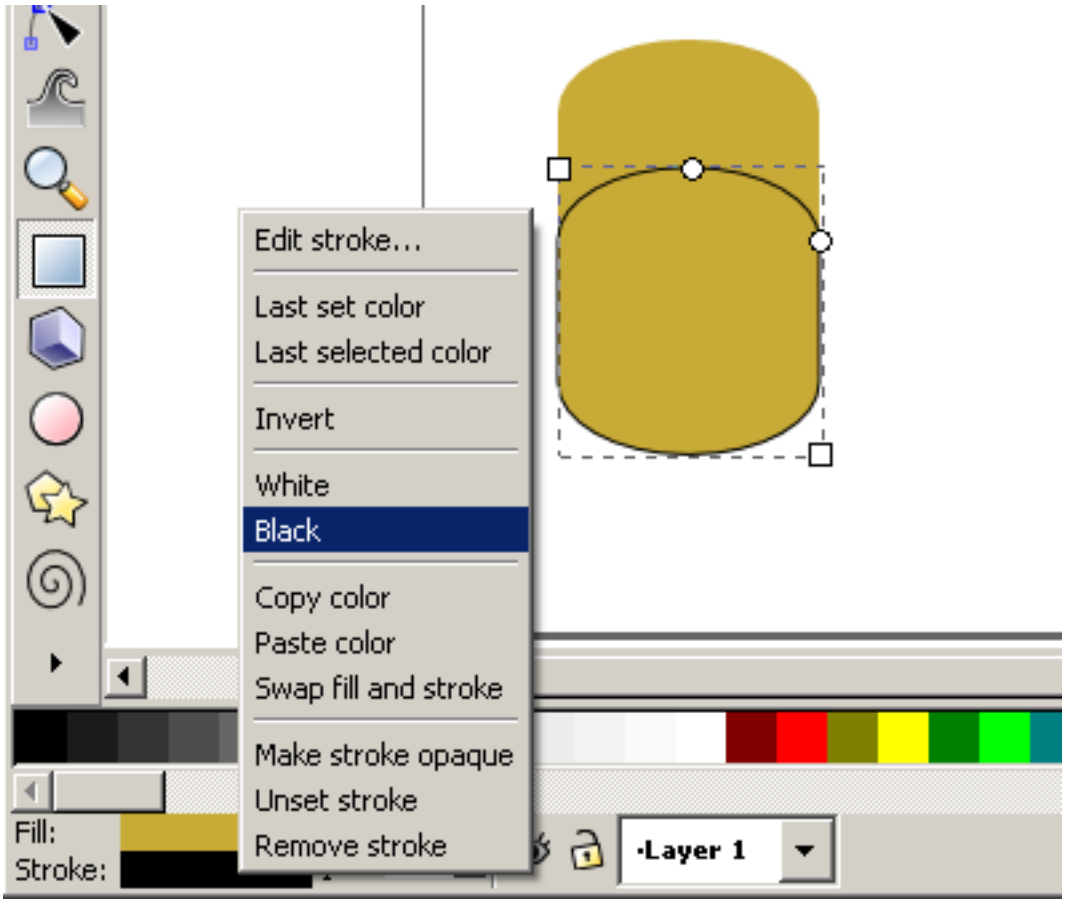
Step 5. When the black object is highlighted, press 'g' to get Gradient tool, then drag mouse across the object from bottom to top. If the shadow effect is not satisfactory, slowly adjust it by dragging the two small circles.



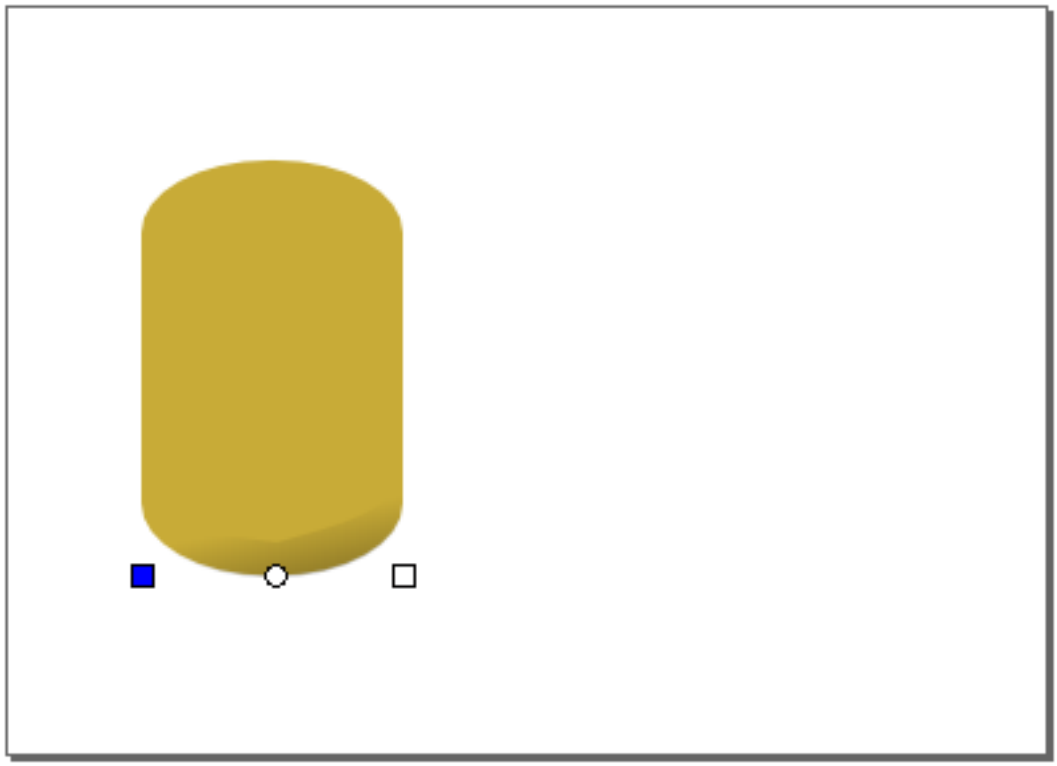
Step 6. Move the top object back to the original position, press page down to ensure shadow is on top, and tank body is at the bottom.



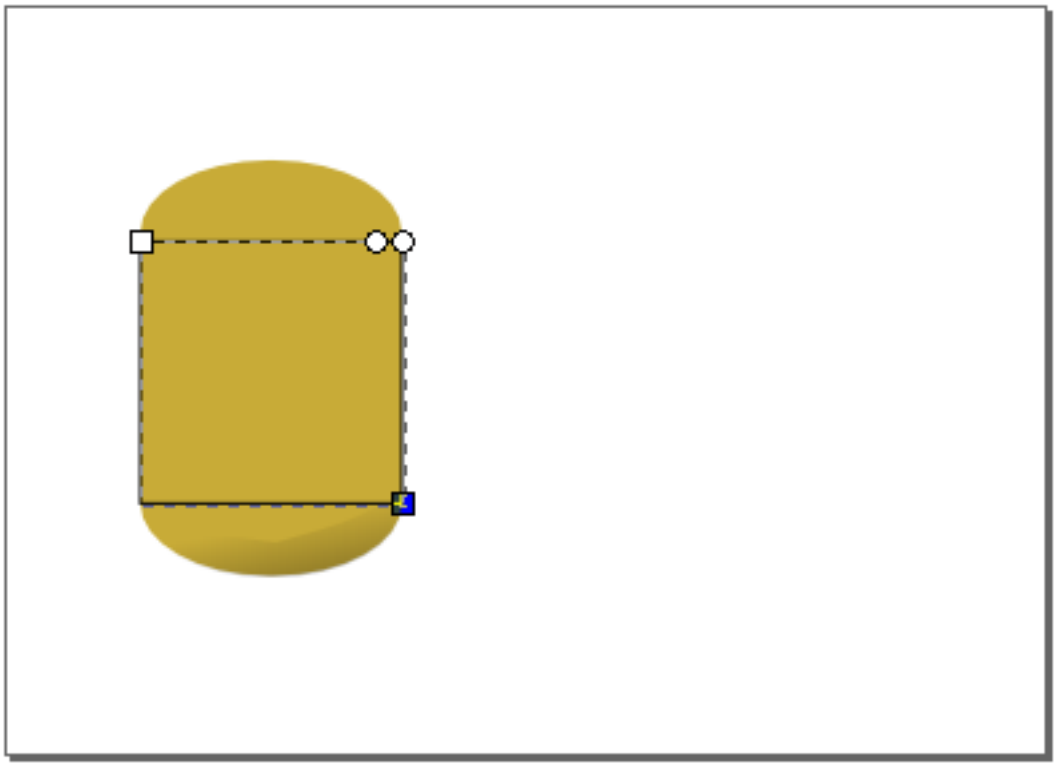
Step 7. Press Ctrl + D again to duplicate one more tank body.



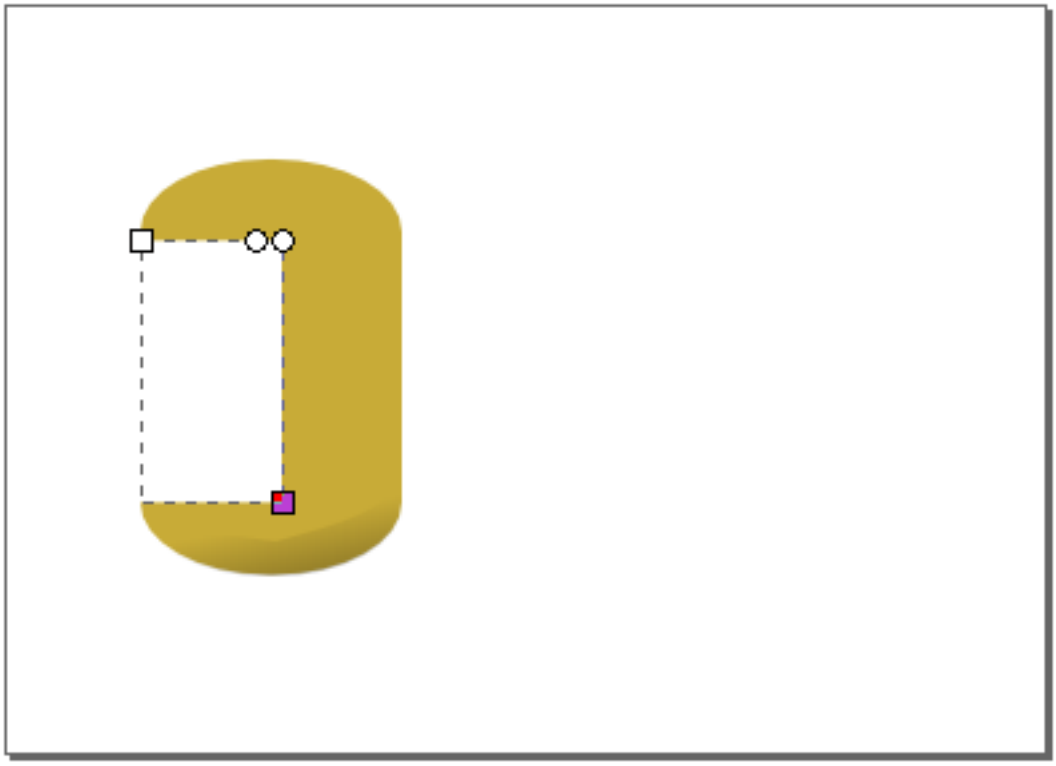
Step 8. For visibility reason, turn the stroke back on by using short cut: right click on 'Stroke' pane at bottom left and select 'Black'. Then resize the newly duplicated body.



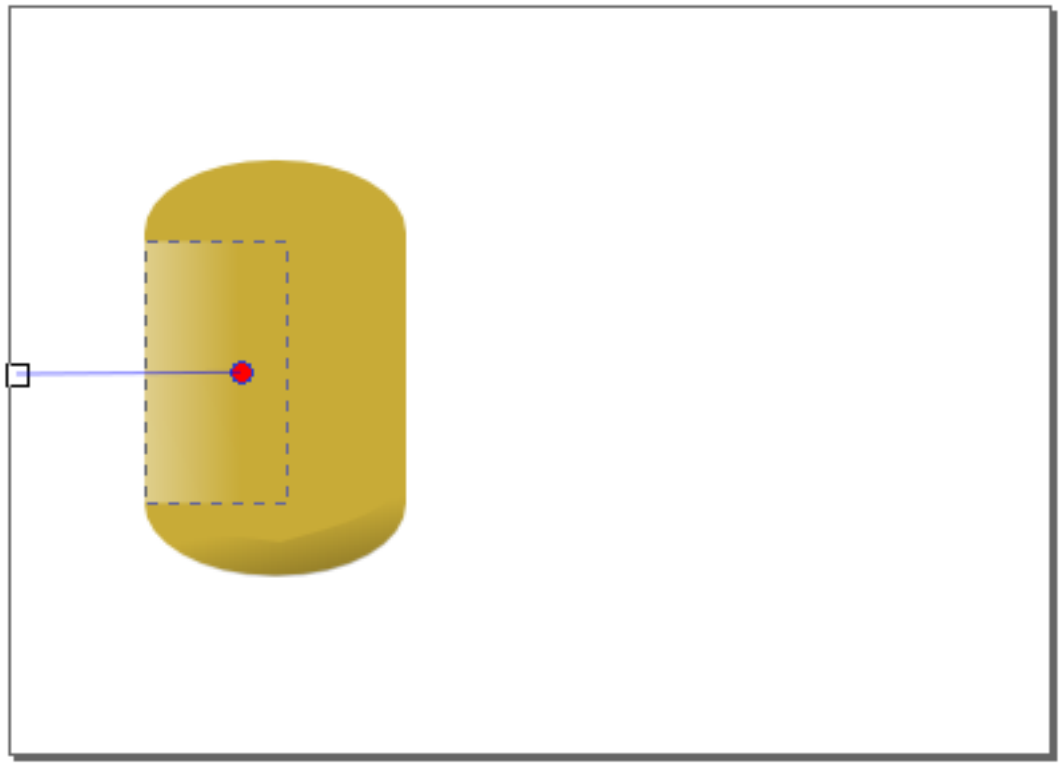
Step 9. Here demonstrate a trick to remove the round corner that inheritted from previously customized shape. Click on the small square circle and drag all the way down until the tank body appear flat as shown.



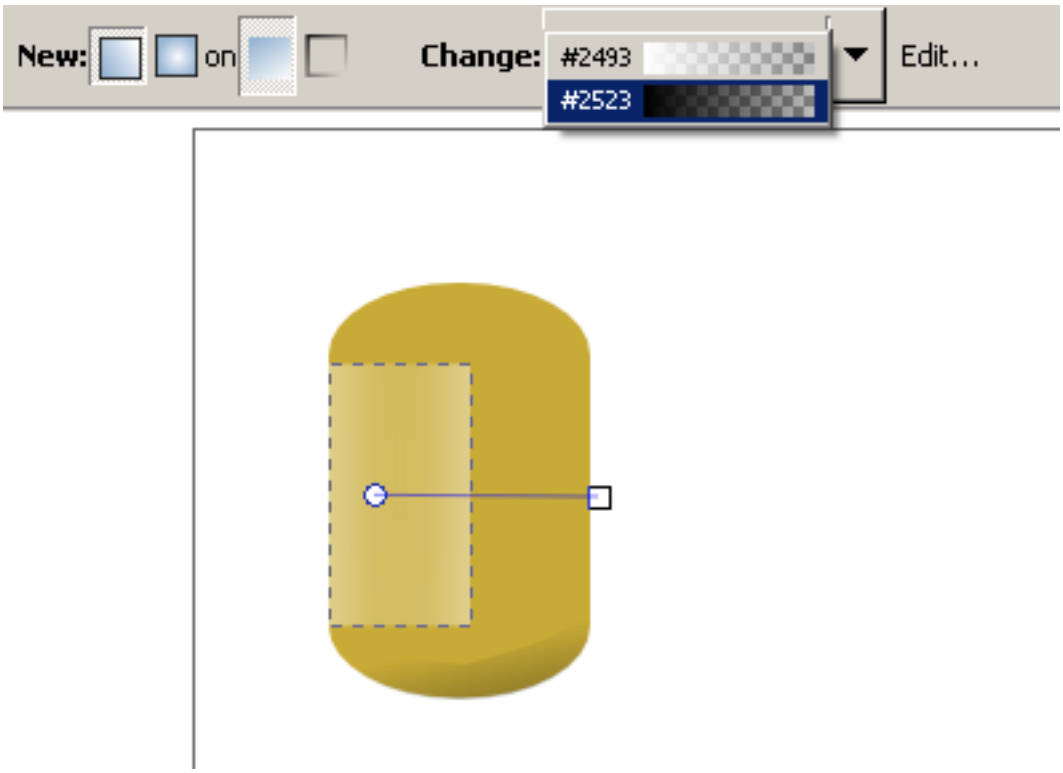
Step 10. Drag the small square circle up again and the round corner will be removed, resize the new rectangle so that it appear in the middle.



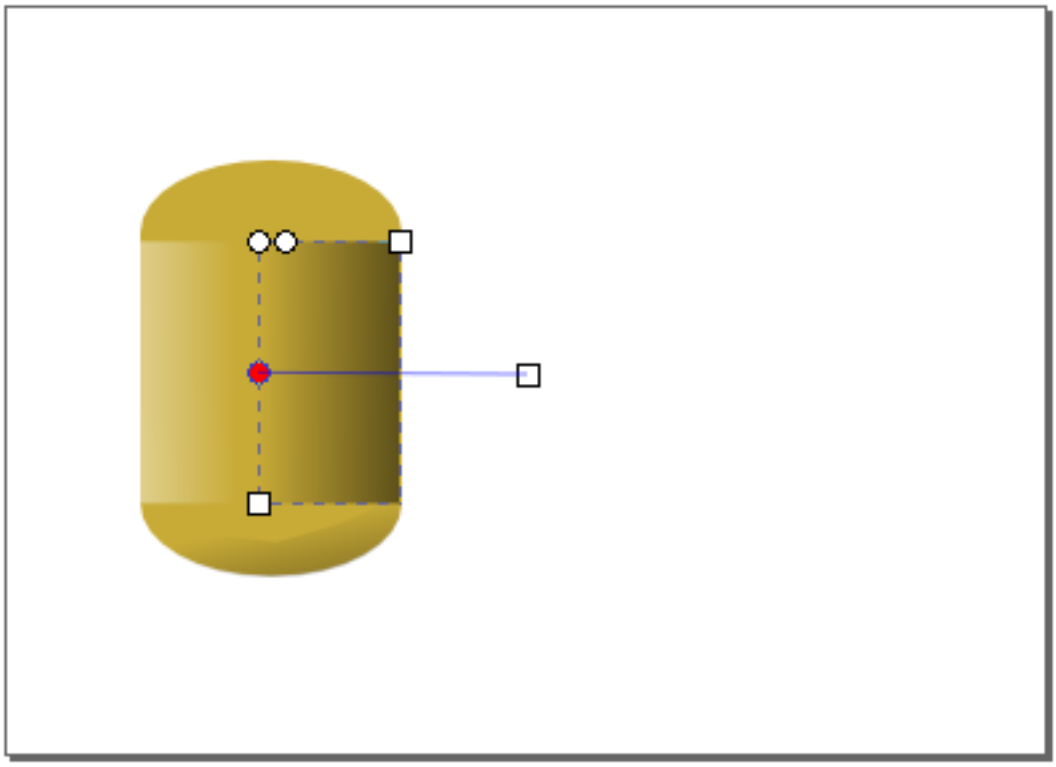
Step 11. Further reduce the size and make it about 1/3 of the tank body. Set the color to white and remove its stroke.



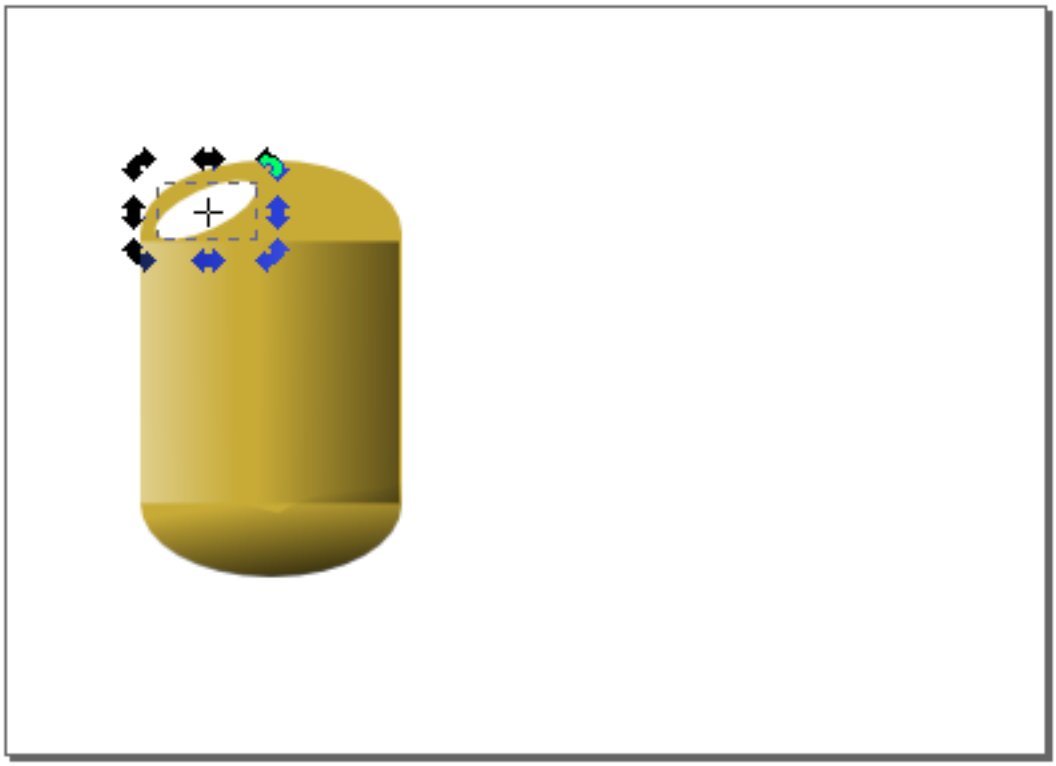
Step 12. Press 'g' to turn on gradient mode and drag across the white rentagle to create bright light source effect.



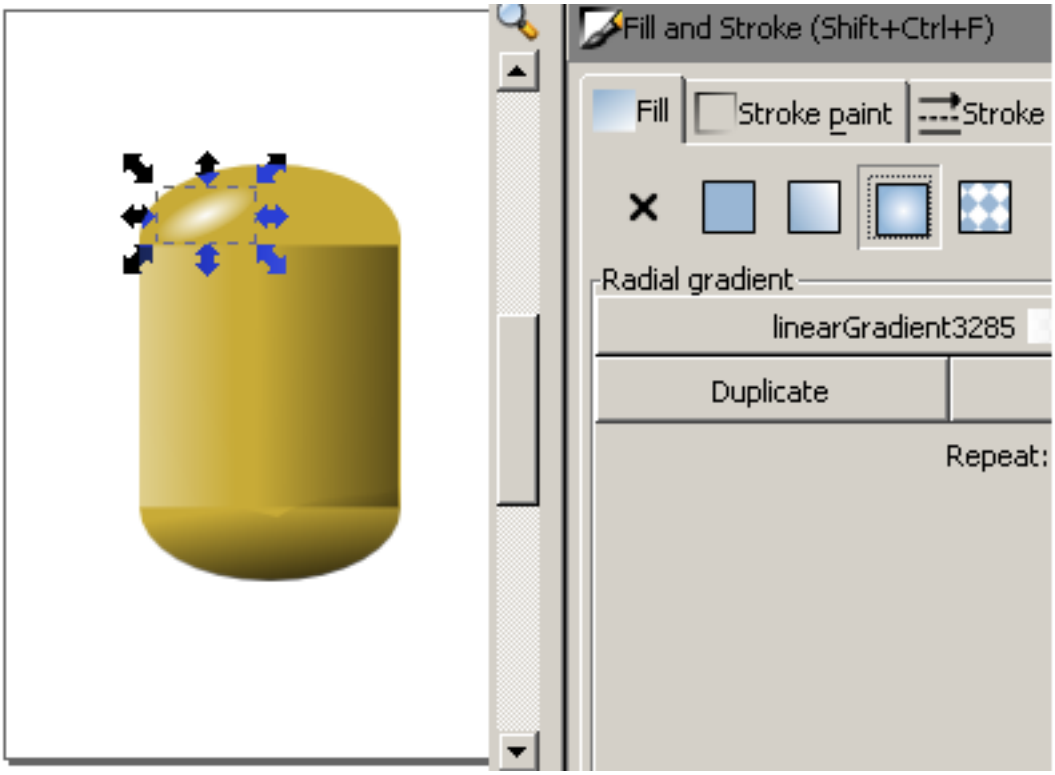
Step 13. Press Ctrl + D to duplicate one more gradient, press 'h' to flip horizontally. Then press Edit Gradient as shown and choose black for shadow effect.



Step 14. Move the shadow to the right, double click to fine tune the effect as desired.



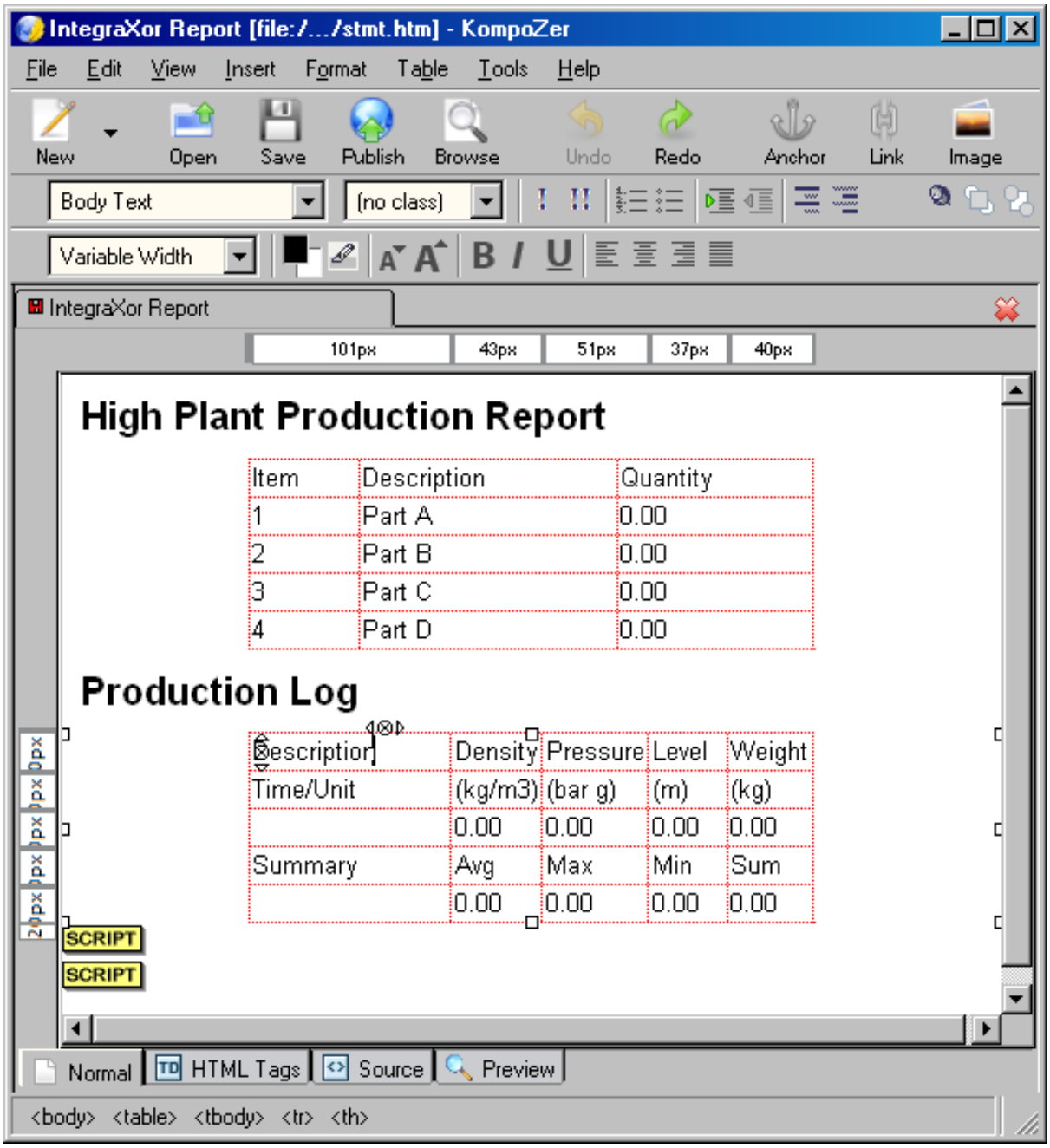
Step 15. Draw a white ellipse on the cover, click once then slightly tilt it to create desired direction of light source.



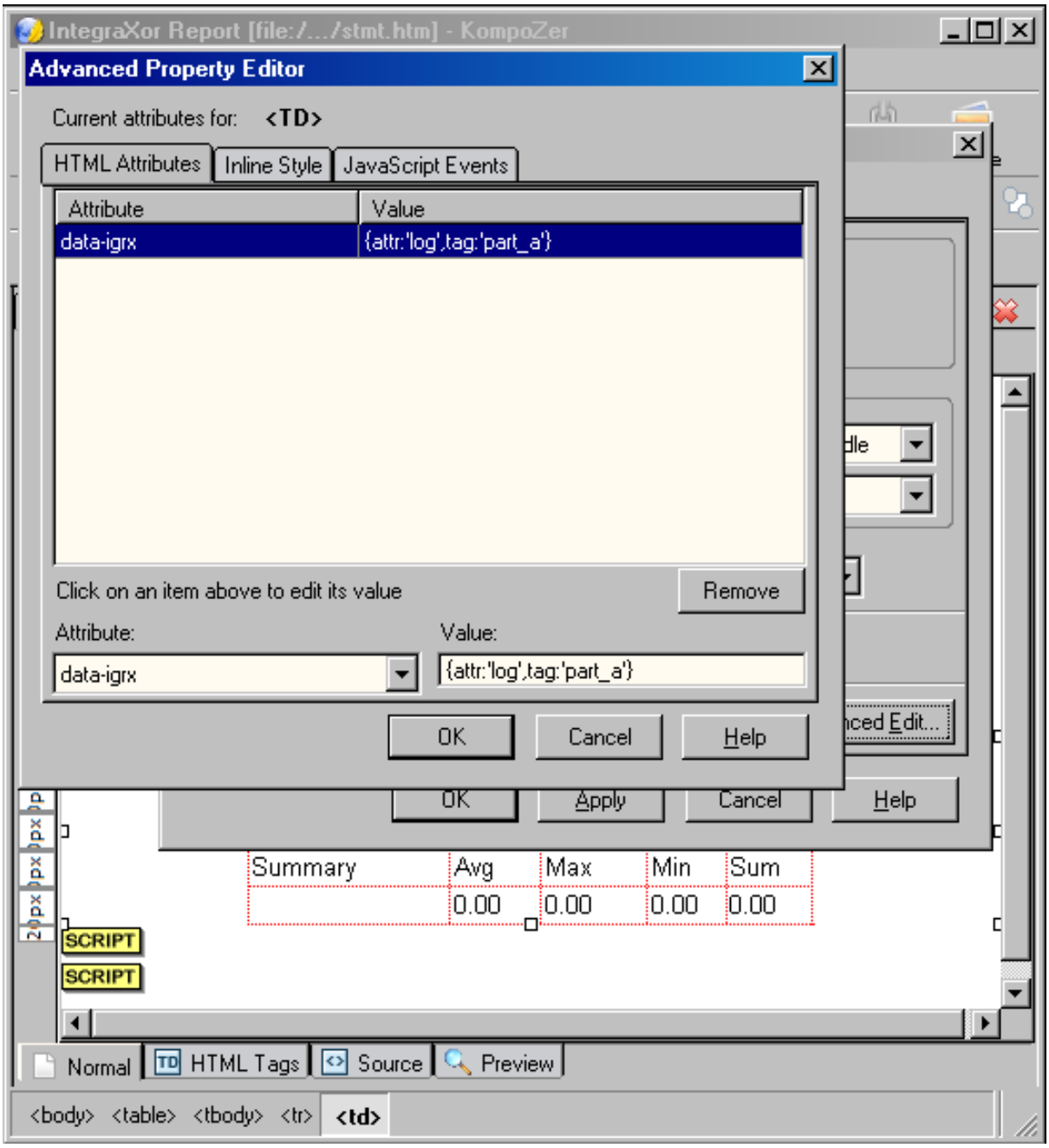
Step 16. Call out the Fill properties, press 'Radial Gradient' to create round surface shiny effect. Again adjust the size and gradient strength to desired result and complete the tank.

12.8. Working with Kompozer

Some engineers prefer to use Visual Editing Tool instead of hand code. Free and open source KompoZer [http://portableapps.com/apps/development/nvu_portable] could be one of the light weight editor for consideration.


IntegraXor Report [file:/.../stmt.htm] - KompoZer
File Edit View Insert Format Table Tools Help
New Open Save Publish Browse Undo Redo Anchor Link Image
Body Text
(no class)
Variable Width
IntegraXor Report
101px 43px 51px 37px 40px
High Plant Production Report
Item Description Quantity
1 Part A 0.00
2 Part B 0.00
3 Part C 0.00
4 Part D 0.00
Production Log
Description Density Pressure Level Weight
Time/Unit (kg/m3) (bar g) (m) (kg)
0.00 0.00 0.00 0.00
Summary Avg Max Min Sum
0.00 0.00 0.00 0.00
SCRIPT
SCRIPT
Normal HTML Tags Source Preview
<body> <table> <tbody> <tr> <th>

Using external visual editor for table editing



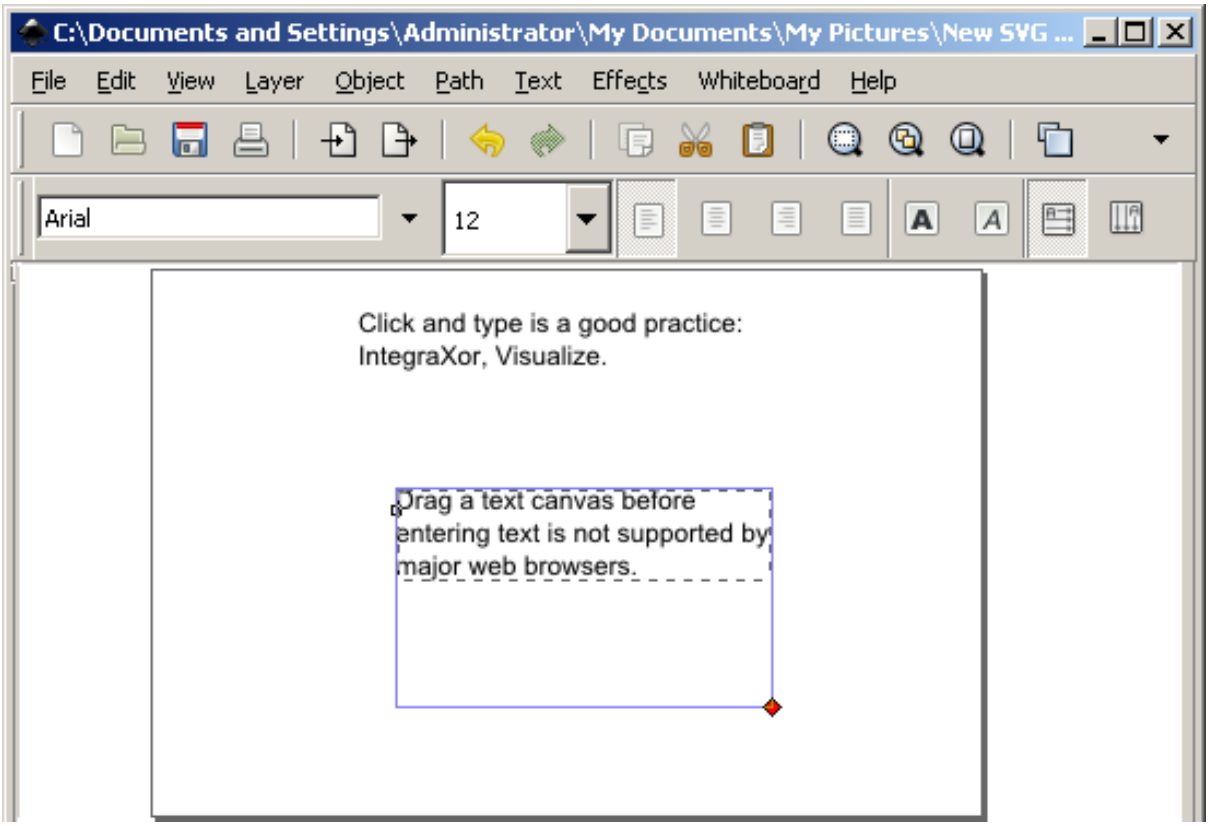
Entering logging attribute via Advance Properties Editor

Chapter 13. Known Issues

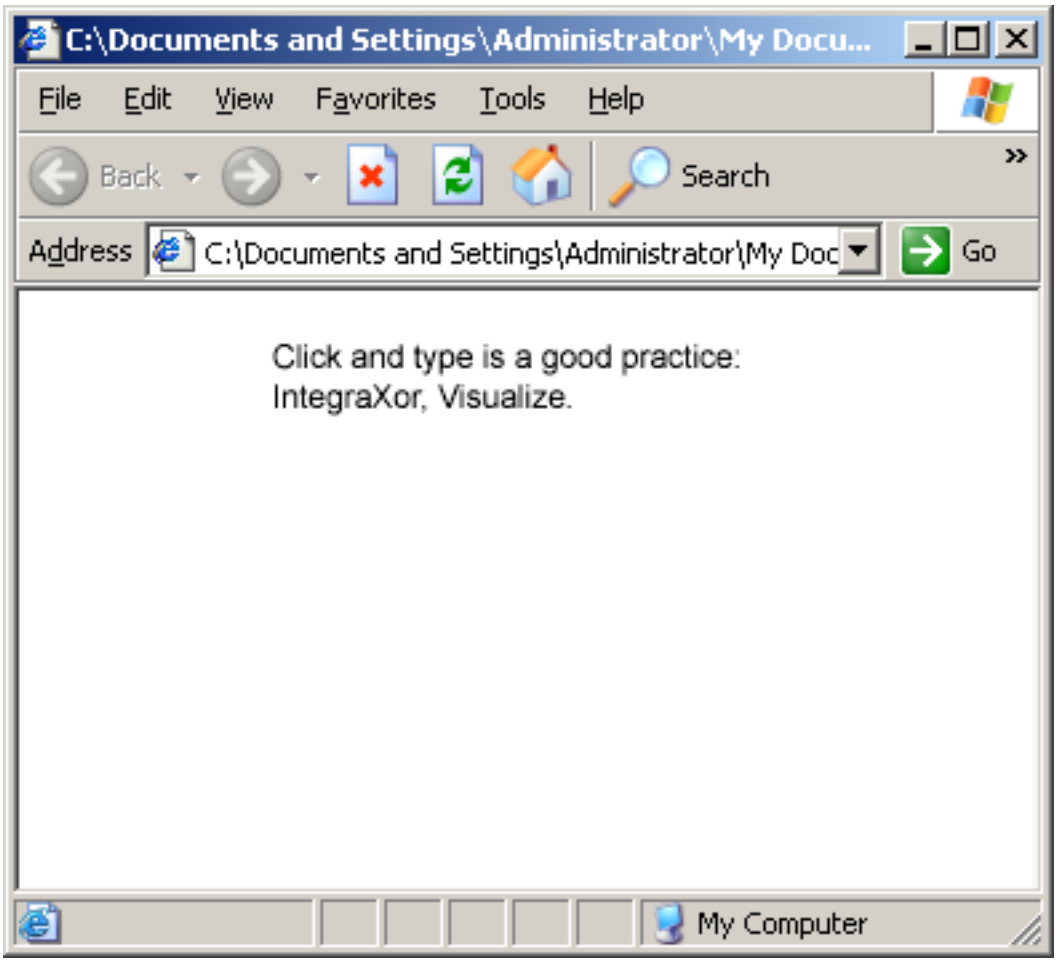
This is a chapter describing temporary known issues, especially those that could not be fixed in short time frame for current release 4.1.4363.0 dated 20 Sep 2013. Beginner who decided to use this product to start off a project should scan through this chapter, but not needed for those who is still exloring and trying. Please check the installed version before reading to ensure the problem described here is still relevant. This chapter will be heavily modified and updated when a problem is found or fixed.

13.1. Inkscape SAGE

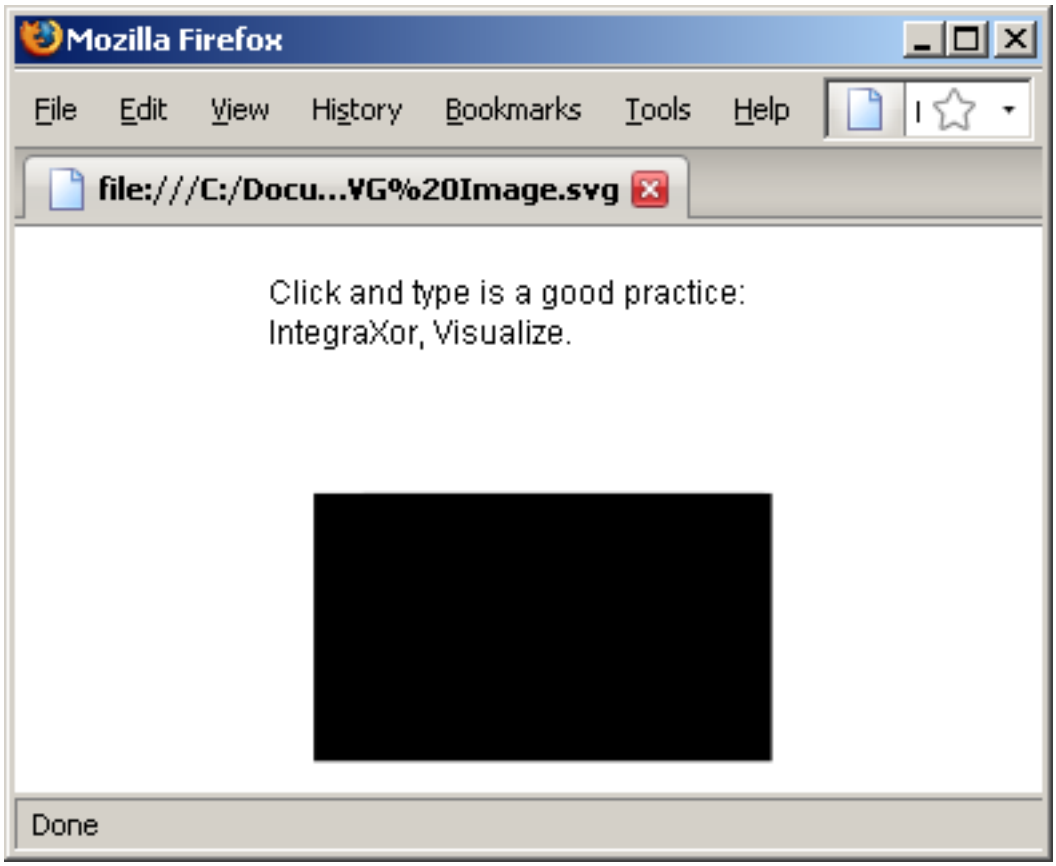
There are two ways of entering text in SVG, one way is simply click and type, the other way is create a text canvas to type so that text could flow within the 'edit box'. Some new users tends to enter text by first dragging to create a text canvas and type within the canvas. Unfortunately text canvas is not supported by major text browsers that running on Windows as of time of writing. So engineer must avoid to use text canvas since none of the major browser support it.


C:\Documents and Settings\Administrator\My Documents\My Pictures\New SVG ...
File Edit View Layer Object Path Text Effects Whiteboard Help
Arial
12
Click and type is a good practice:
IntegraXor, Visualize.
Drag a text canvas before
entering text is not supported by
major web browsers.

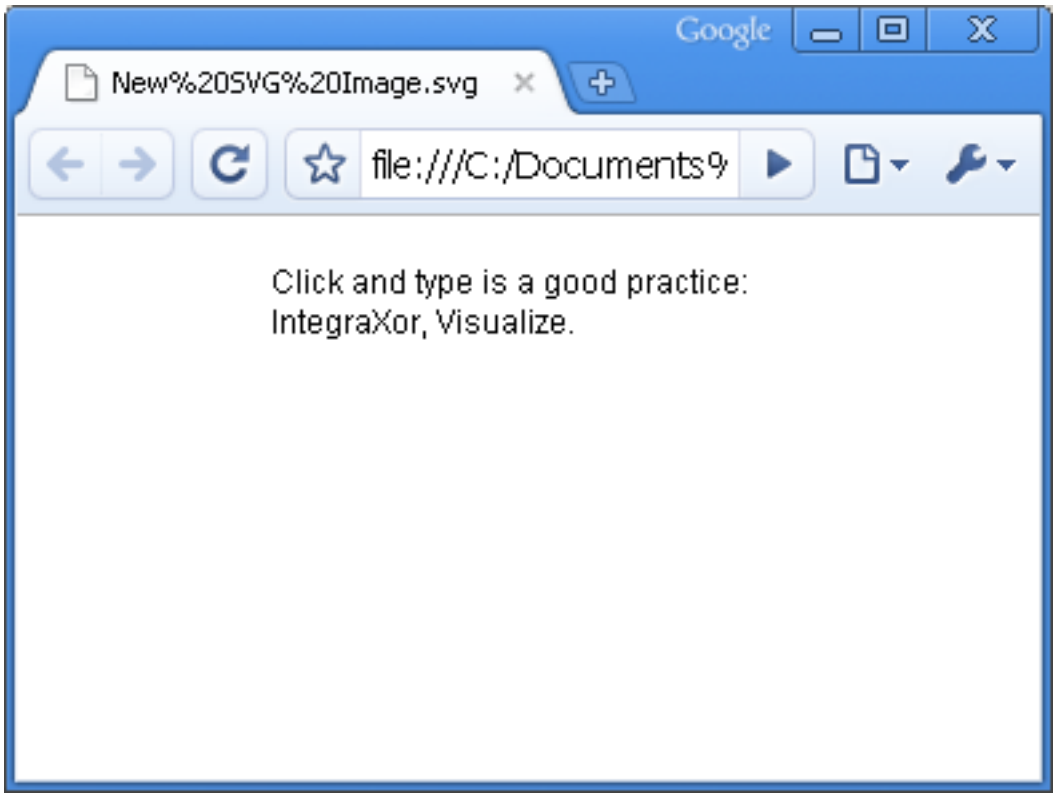
Inkscape supports two types of text entering mechanism, but browsers only display text created from Click & Type.



ASV for I.E. shows blank for text created via canvas.



Firefox 3.0 shows black rectangle on text canvas.



Google Chrome shows blank on text canvas.

Warning

As a designer level graphic editing tool, Inkscape may still crash in certain scenarios and has its own issues [http://wiki.inkscape.org/wiki/index.php/Release_Notes/0.48#Known_Issues]. Although this might be rare for standard usage, users are advised to press Ctrl+S to *save* as frequent as possible to avoid loosing any work.

13.1.1. Miscellaneous

Script	SAGE didn't handle the escape character back-slash (\) well. If back-slash is used in script and clicking [Apply] button will generate several slashes behind. The work around is to delete any additional slash, avoid to use [Apppy] button but simply click [OK] only.
Raster Image Scaling	Browsers cannot handle non proportional scaling for raster image [http://www.integraxor.com/blog/browsers-always-preserves-aspect-ratio-when-resizing-raster-image-in-svg] like *.jpg or *.png. Engineer will normally bump into this problem when uses these images as background like factory floor or building plan, then adjust the size slightly withtout noticing it's out of proportion. When engineer start drawing SVG graphic on top of it which looks absolutely aligned and perfect, the outcome is they looks out of place slightly here and there during run-time in browser. To solve this problem, remove the scaling of the raster image, then rescale the image to desired position by holding down Ctrl button. Another workaround is to draw additional dummy SVG object behind the raster image, and then group them together before scaling.

13.2. Project Editor

Copy & paste by row	Copy one or more rows for pasting requires user to highlight the targeted row in full before pasting, but not on the individual cell. Otherwise the content will not match to the original column. Workaround for this fool proof issue is to always right click on the row number before pasting.
Text Editor	Text editor will have cosmetic error with Windows EOL format which has dual indicator, i.e. CR & LF. First known issue is user need to press Delete/Backspace twice when removing line break. Second known issue is Find-All will have wrong indication of line number. The workaround is to convert EOL into Unix/OXS format. For Notepad++ user could find this command at Menu\Edit\EOL conversion.
HTML Editor	Clicking buttons at the bottom for Select Tag/Content in HTML/Text editor will not yield any practicle outcome. For workaround, use mouse to manually highlight in order to select. (3.72.4230.1)
Undo changes	Current version of Project Editor does not support undo for changes. Please save the working project as frequent as possible, so user could discard the undesired changes by closing the current pane (Ctrl+F4 / Ctrl+W).

Upgrade from old manual menu

Upgrading old projects that has no `menu.js` in project directory will trigger a pop up "Error running menu.js, please check again your script.", user must manually copy a `menu.js` from any template into the old project in order to proceed.

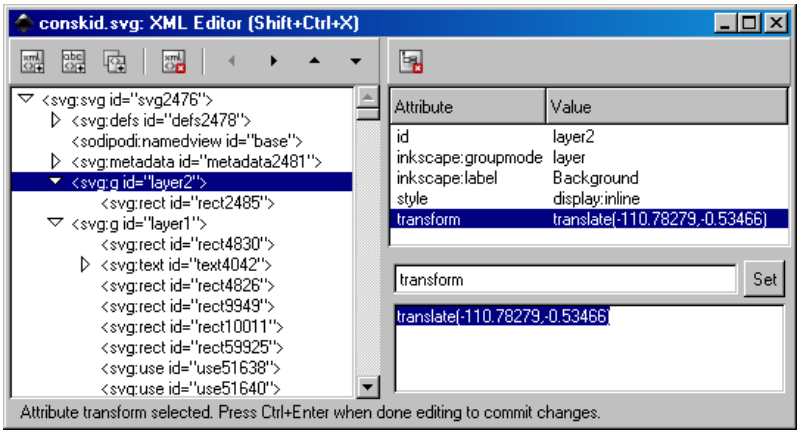
13.3. Front-end

Animation Engine

A SVG graphic layer must *not* be named as 'image' in Inkscape SAGE as it will be conflicting with the system keywords during runtime.

Auto Fit is off-centered

Any SVG layer must *not* be manually "transformed", otherwise the SVG mimic will appeared out of center during runtime.



Entering any transform setting will results in off-centered screen during runtime. Value for 'transform' at right bottom shall be deleted.

SAGE - Clone

Do not use Clone for graphic object that need to be animated if Google Chrome is one of the targeted browser. Animation behaviour for GC on cloned graphic object is unpredictable.

Firefox - Cache

Firefox is a popular development platform, however, engineer shall be aware that most browsers including Firefox will try to cache browsing data. If update can't be seen after making changes, simply use another browser to double confirm. Also try to press Ctrl+F5 for forceful refresh. Otherwise please learn how to clear cache [http://support.mozilla.com/en-US/kb/How%20to%20clear%20the%20cache].

During development, engineer could view the page being developed directly by entering the full path: `http://localhost:7131/job/page.htm?YYMMDD-hhmm` Pay attention to the *questionmark* + *Timestamp* added to the end. This is known as cache-buster trick whereby the timestamp need to be changed every time a complete refresh is expected.

Chrome - Cache

Google Chrome has an Incognito mode [https://support.google.com/chrome/answer/95464?hl=en] that you can avoid any cache issue.

Google Chrome - Log in/out cache

Google Chrome remembers passwords even though user has logged out,

	therefore does not prompt to login until the browser is closed. This is a known issue [http://code.google.com/p/chromium/issues/detail?id=5497] for Google Chrome.
Browsers always preserves aspect ratio when resizing [http://www.integraxor.com/blog/browsers-always-preserve-aspect-ratio-when-resizing-svg]	To circumvent this browsers' problem is to group the raster image with other SVG elements and then scale it together. Otherwise the best thing to do is to avoid scaling or at least scale it in proportion by holding down Ctrl key, in ensuring WYSIWYG in browsers' display.

13.4. Miscellaneous

Report - Engineering Unit	Report parser does not support HTML entities like °, ³ and etc mentioned. Please avoid to use them till this notice is removed.

Index

A

alarm
  configuration, 45
  deadband, 47
  script, 48
animation
  bar, 75
  color, 76
  get, 72
  opacity (transparency), 80
  open window page, 81
  pop up, 90
  rotate, 82
  script, 89
  set, 74
  slider position, 86
  Text, 9l, 92
  tooltips, 85
  zoom, 87
Animation
  text alignment, 73
architecture, 8

C

communication
  device, 36
  port, 36
  troubleshoot, 117
configuration
  project editor, 33
  sequence, 33

D

database
  configuration, 42
  create table, 15l
Development
  concurrent, 150
device
  configuration, 36
  driver, 127
driver
  EtherNet/IP, 130
  IEC60870-5-101/104, 138
  modbus, 127
  Profinet, 134
  SNMP, 136
  SQL, 131

F

front-end
  alarm viewer, 60
  expression, 69
  html tabulate data, 59
  introduction, 55
  language, 68
  number formating, 58
  print, 59
  trend, 62

G

general
  configuration, 34

I

inkscape
  gradient, 163
  graphic editing, 153
  introduction, 18
  layer, 154
  path, 156
  SAGE, 71
  sizing & background, 154
installer
  installation, 3, 5

L

license
  registration & authorization, 27

M

Modbus
  configuration, 36
  error message, 124
  protocol, 122

N

Network Redundancy
  Configuration, 142
  Introduction, 141

O

Omron - Fins
  configuration, 36
OPC
  configuration, 36
  DCOM registry, 126

P

port
  configuration, 36

project
backup, 149
migration, 150
project editor
getting started, 11
tag auto complete, 17

R

Redundancy
Introduction, 141
Report
Introduction, 95
requirements, 2

S

script
accessing tag, 57
ActiveX scripting, 52
additional functions, 49
advanced scripting, 49
configuration, 48
database functions, 50
debugging functions, 5l
JSON, 58
security
additional, 114
introduction, 111
protection, 113
user configuration, 112
server
user interface, 25
Server Redundancy
Configuration, 145
Introduction, 143

T

tag
configuration, 38
scaling/expression, 40
template
getting started, 17
project folder, 149
update, 150
timer
configuration, 35
tools
utilities, 150
Tools
Visual HTML Editor, 172
traffic
analysis, 120
tutorial
getting started, 11

U

URL
address, 45
user security login
configuration, 44